

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-258977

(43) 公開日 平成9年(1997)10月3日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	5 5 0		G 0 6 F 9/06	5 5 0 A
				5 5 0 D
12/14	3 2 0		12/14	3 2 0 B
G 0 9 C 1/00	6 3 0	7259-5 J	G 0 9 C 1/00	6 3 0 B
	6 6 0	7259-5 J		6 6 0 D

審査請求 未請求 請求項の数19 O L (全 42 頁) 最終頁に続く

(21) 出願番号 特願平8-165637

(22) 出願日 平成8年(1996)6月26日

(31) 優先権主張番号 特願平8-5559

(32) 優先日 平8(1996)1月17日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂二丁目17番22号

(72) 発明者 田口 正弘

神奈川県足柄上郡中井町境430 グリーン  
テクなかい 富士ゼロックス株式会社内

(72) 発明者 河野 健二

神奈川県足柄上郡中井町境430 グリーン  
テクなかい 富士ゼロックス株式会社内

(72) 発明者 齊藤 和雄

神奈川県足柄上郡中井町境430 グリーン  
テクなかい 富士ゼロックス株式会社内

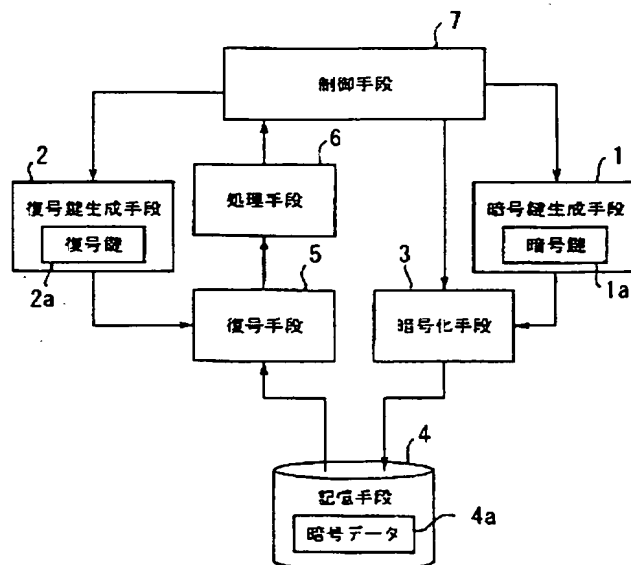
(74) 代理人 弁理士 服部 毅藏

(54) 【発明の名称】 ソフトウェアの保護機能付き情報処理装置

(57) 【要約】

【課題】 システムのメモリの管理方法に依存せずに高い暗号強度を得る。

【解決手段】 暗号化対象データが入力されると、暗号鍵生成手段1が、その暗号化対象データの属性に応じて暗号鍵1aを生成する。この暗号鍵1aを用いて、暗号化手段3が暗号化対象データを暗号化し、暗号データ4aが記憶手段4に格納される。そして、暗号データ4aの処理要求があると、復号鍵生成手段2が暗号データ4aの属性に応じて復号鍵2aを生成する。復号手段5は、その復号鍵2aを用いて、暗号データ4aを復号する。復号されたデータは、処理手段6により処理される。制御手段7は、処理手段6により処理されたデータを暗号化対象データとして暗号化手段3に対して出力する。これにより、メモリ管理方法に依存せずに、属性毎に異なる鍵を使用し暗号強度を高めることができる。



**【特許請求の範囲】**

【請求項1】 データの暗号化と復号とを行いながら動作するソフトウェアの保護機能付き情報処理装置において、  
命令とデータを含む暗号化対象データの属性に応じて暗号鍵を生成する暗号鍵生成手段と、  
暗号データの属性に応じて復号鍵を生成する復号鍵生成手段と、  
前記暗号鍵生成手段が生成した暗号鍵を用いて、暗号化対象データを暗号化する暗号化手段と、  
前記暗号化手段が暗号化した暗号データを格納する記憶手段と、  
前記記憶手段に記憶されている暗号データを、前記復号鍵生成手段が生成した復号鍵を用いて復号する復号手段と、  
前記復号手段に直接接続されており、前記復号手段が復号したデータに含まれる命令に従って、復号したデータを処理する処理手段と、  
前記処理手段の処理終了後のデータを直接に、暗号化対象データとして前記暗号化手段へ出力する制御手段と、  
を有することを特徴とするソフトウェアの保護機能付き情報処理装置。

【請求項2】 前記暗号鍵生成手段は、暗号化対象データのアドレスまたはアドレス領域に応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データのアドレスまたはアドレス領域に応じて復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項3】 前記記憶手段は、仮想記憶方式により記憶空間が管理されており、  
前記暗号鍵生成手段は、暗号化対象データの仮想アドレスまたは仮想アドレス領域に応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの仮想アドレスまたは仮想アドレス領域に応じて復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項4】 前記記憶手段は、ページ方式の仮想記憶方式により記憶空間が管理されており、  
前記暗号鍵生成手段は、暗号化対象データの属するページに応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの属するページに応じて復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項5】 前記記憶手段は、セグメント方式の仮想記憶方式により記憶空間が管理されており、  
前記暗号鍵生成手段は、暗号化対象データの属するセグメントに応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの属するセグメント

に応じて復号鍵を生成する、

ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項6】 前記記憶手段は、多重仮想記憶方式により記憶空間が管理されており、  
前記暗号鍵生成手段は、暗号化対象データの属する仮想記憶空間に応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの属する仮想記憶空間に応じて復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項7】 前記処理手段は、プロセス毎に処理を行っており、  
前記暗号鍵生成手段は、暗号化対象データの属するプロセスに応じて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの属するプロセスに応じて、復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項8】 前記暗号鍵生成手段は、暗号化対象データの属性として、暗号化対象データそのものを用いて暗号鍵を生成し、  
前記復号鍵生成手段は、暗号データの属性として、暗号データそのものを用いて復号鍵を生成する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項9】 前記暗号鍵生成手段、前記復号鍵生成手段、前記暗号化手段、前記復号手段、前記処理手段、及び前記制御手段を囲む包囲体に対して外部から物理的な作用を受けると、各処理機能を司るデータを消去する安全保護手段を、  
さらに具備し、

前記暗号化手段は、前記安全保護手段内で処理されたデータを暗号化して前記包囲体の外部に出力し、  
前記復号手段は、前記記憶手段に格納されている暗号データを前記包囲体の内部に取り込んでから復号する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項10】 予め復号鍵と復号アルゴリズムとが特定されている暗号データが入力されると、入力された暗号データを復号し、暗号化対象データとして前記暗号化手段に入力する流通データ復号手段を、  
さらに具備することを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項11】 前記暗号鍵生成手段は、複数の暗号鍵を暗号鍵群として管理しており、複数の暗号化対象データの集まりの属性に応じて暗号鍵群を選択し、  
前記復号鍵生成手段は、複数の復号鍵を復号鍵群として管理しており、複数の暗号データの集まりの属性に応じて復号鍵群を選択し、

前記暗号化手段は、複数の暗号化対象データの集まりの個々の暗号化対象データを、前記暗号鍵生成手段が選択した暗号鍵群の個々の暗号鍵を用いて暗号化し、  
前記復号手段は、複数の暗号データの集まりの個々の暗号データを、前記暗号鍵生成手段が選択した復号鍵群の個々の復号鍵を用いて復号する、  
ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項12】 暗号化した際の暗号鍵を変更する必要がある再暗号化対象データの再暗号化指令を出力する再暗号化指令手段を、

さらに具備し、

前記暗号鍵生成手段は、前記再暗号化指令を受け取ると、再暗号化対象データの属性に対応させる暗号鍵を変更し、変更後の暗号鍵を生成し、

前記復号鍵生成手段は、前記再暗号化指令を受け取ると、再暗号化対象データの属性に応じた復号鍵を生成するとともに、再暗号化対象データの全てが復号された時点で、再暗号化対象データの属性に対応させる復号鍵を変更し、

前記復号手段は、再暗号化対象データを、前記復号鍵生成手段が生成した復号鍵を用いて復号し、

前記暗号化手段は、復号された再暗号化対象データを、前記暗号データ生成手段が生成した暗号鍵を用いて暗号化する、

ことを特徴とする請求項1記載のソフトウェアの保護機能付き情報処理装置。

【請求項13】 前記再暗号化指令手段は、暗号鍵の有効期限を管理しており、任意の暗号鍵の有効期限がきれると、前記有効期限がきれた暗号鍵により暗号化された暗号データを再暗号化対象データとして、再暗号化指令を出力することを特徴とする請求項12記載のソフトウェアの保護機能付き情報処理装置。

【請求項14】 データの暗号化と復号とを行いながら動作するソフトウェア保護機能付き情報処理装置において、

命令とデータを含む暗号化対象データの属性に応じて暗号鍵を生成する暗号鍵生成回路と、

暗号データの属性に応じて復号鍵を生成する復号鍵生成回路と、

前記暗号鍵生成回路が生成した暗号鍵を用いて、暗号化対象データを暗号化する暗号化回路と、

前記暗号化回路が暗号化した暗号データを格納するメモリと、

前記メモリに記憶されている暗号データを、前記復号鍵生成回路が生成した復号鍵を用いて復号する復号回路と、

前記復号回路に直接接続されており、前記復号回路が復号したデータに含まれる命令に従って、復号したデータを処理するMPUと、

前記MPUの処理終了後のデータを直接に、暗号化対象データとして前記暗号化回路へ出力する制御回路と、  
を有することを特徴とするソフトウェアの保護機能付き情報処理装置。

【請求項15】 前記暗号鍵生成回路は、暗号化対象データのアドレスまたはアドレス領域に応じて暗号鍵を生成し、

前記復号鍵生成回路は、暗号データのアドレスまたはアドレス領域に応じて復号鍵を生成する、

ことを特徴とする請求項14記載のソフトウェアの保護機能付き情報処理装置。

【請求項16】 データの暗号化と復号とを行いながら動作するソフトウェアの保護機能付き情報処理装置において、

暗号化又は復号の際に使用される複数の鍵を供給する鍵供給手段と、

暗号化又は復号の際のアルゴリズムを表現した複数のアルゴリズム情報を供給するアルゴリズム供給手段と、  
命令とデータを含む暗号化対象データの暗号化の際には、供給されている前記複数の鍵と前記複数のアルゴリズム情報の中から暗号鍵と暗号化アルゴリズム情報とを選択し、暗号データの復号の際には、供給されている前記複数の鍵と前記複数のアルゴリズムの中から復号鍵と復号アルゴリズム情報とを選択する暗号化／復号方式選択手段と、

前記暗号化／復号方式選択手段が選択した暗号鍵と暗号化アルゴリズム情報とを用い、暗号化対象データを暗号化する暗号化手段と、

前記暗号化手段が暗号化した暗号データを格納する記憶手段と、

前記暗号化／復号方式選択手段が選択した復号鍵と復号アルゴリズム情報とを用い、前記記憶手段に記憶されている暗号データを復号する復号手段と、

前記復号手段に直接接続されており、前記復号手段が復号したデータに含まれる命令に従って、復号したデータを処理する処理手段と、

前記処理手段の処理終了後のデータを直接に、暗号化対象データとして前記暗号化手段へ出力する制御手段と、  
を有することを特徴とするソフトウェアの保護機能付き情報処理装置。

【請求項17】 前記暗号化／復号方式選択手段は、暗号化の際に選択すべき暗号鍵と暗号化アルゴリズムとの組み合わせ、及び復号の際に選択すべき復号鍵と復号アルゴリズムとの組み合わせを、所定の規則に従い随時変更する、

ことを特徴とする請求項16記載のソフトウェアの保護機能付き情報処理装置。

【請求項18】 前記アルゴリズム供給手段は、供給するアルゴリズム情報の内容を、所定の規則に従い随時変更する、

ことを特徴とする請求項16記載のソフトウェアの保護機能付き情報処理装置。

【請求項19】 前記アルゴリズム供給手段は、供給しているアルゴリズム情報の使用期限を管理しており、前記使用期限を超えて供給されているアルゴリズム情報を他のアルゴリズム情報に変更する、

ことを特徴とする請求項18記載のソフトウェアの保護機能付き情報処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータ等に取り扱うプログラム及びデータを盗用や改ざん及び不正使用から保護するソフトウェアの保護機能付き情報処理装置に関し、特に汎用性の高いソフトウェアの保護機能付き情報処理装置に関する。

【0002】

【従来の技術】作成したプログラムやデータ（以下、特にことわりのない限りプログラムとデータとを含めて単に「データ」と呼ぶ）を流通させる場合、データを盗用や改ざんあるいは不正使用から保護する必要がある。これまで保護手段としては、データをロム化したり、フロッピー等に記憶しコピープロテクトをかける等の方法がとられてきた。しかし、このような方法ではデータの内容を読み出そうと思えば容易に読み出すことが可能であるためデータはほとんど保護されていない。

【0003】また、他のデータの保護方法としてデータを暗号化して供給し、復号鍵を持ったユーザのみがデータを復号して使用できるようにする技術がある。この方法では、復号された後のデータがメモリや固定ディスク装置などに蓄えられるため、盗用や改ざん及び不正使用の危険を払拭することはできない。

【0004】これらを解決する技術として、復号された後のデータを不正に入手できないようにするために、データを暗号化してメモリ等に格納しておき、それを中央処理装置で実行する時に復号する方式が、特開平2-15034号の「セキュリティ機能付き情報処理装置」に示されている。この方式では、データを保護するために、情報処理装置内部に暗号化装置と復号装置とが設けられており、これにより、ソフトウェアの保護を図っている。以下に、このような情報処理装置について具体的に説明する。

【0005】図32は従来のソフトウェアの保護を図った情報処理装置のブロック図である。図32に示した情報処理装置は、中央処理装置210、記憶装置240、入力装置220、出力装置230、及び鍵入力装置250を備えている。さらに、中央処理装置210は、内部に演算部212、制御部211、暗号化・復号部213、及び鍵格納部214を備えている。

【0006】中央処理装置210は、情報処理装置の中心部分として機能し、データの演算や他の装置の制御等

を行う。記憶装置240は、データが格納される装置であり、中央処理装置210からの制御によって中央処理装置210内の暗号化・復号部213とデータの授受を行う。

【0007】入力装置220は、中央処理装置210からの制御で情報処理装置の外部からのデータを受け取る。出力装置230は、中央処理装置210からの制御で情報処理装置のデータを外部に出力する。鍵入力装置250は、中央処理装置210の暗号化や復号を行うのに必要な鍵をセットする。記憶装置240は、データが格納される装置であり、中央処理装置210からの制御によって中央処理装置210内の暗号化・復号部213とデータの授受を行う。

【0008】中央処理装置210内の演算部212は入力装置220や記憶装置240から与えられたデータに対して算術演算や論理演算を行う。制御部211は記憶装置240からの命令を解釈し情報処理装置全体の制御を行う。鍵格納部214は、鍵入力装置250がセットした鍵を格納する。暗号化・復号部213は記憶装置240と演算部212との間にあり、暗号化されている記憶装置240上の命令及びデータを、鍵格納部214内の鍵を用いて演算部212が解釈できるように復号するとともに、演算部212で演算された結果を記憶装置240に書き込む際に、鍵格納部214内の鍵を用いて暗号化し記憶装置240に格納する。

【0009】このような構成により、記憶装置240に格納されるデータを常に暗号化しておくことができる。そのため、記憶装置240内のデータを盗用してもその内容を解釈することは困難となり、データの秘匿性を高めることができる。

【0010】ところで、このようなソフトウェアの保護機能が付いた情報処理装置においては、暗号化して格納されたデータを実行時に逐次復号して実行しなければならない。そのため、復号する際のオーバーヘッドを考慮して計算量が少ない比較的簡単なアルゴリズムの暗号化方式（例えばXOR等）を用いなければならない。結果として暗号強度が低くなるという問題がある。従って、比較的簡単なアルゴリズムの暗号化方式でも暗号強度を出来るだけ高める必要がある。

【0011】そのため、情報処理装置ごとに暗号化の方法を変えて非公開にする努力がなされている。ただし、このような方法を用いるとデータの互換性が大きく失われてしまうという新たな問題が発生する。しかも、プログラム内の命令コードの出現頻度や暗号化されたプログラムと装置の動作の対応などから、暗号化アルゴリズムや暗号鍵が類推できるので、装置単位で見ると必ずしも暗号強度が高くなったとは言えない。

【0012】そこで、少ない計算量で高い暗号強度を得られるような、他の手法がいくつか考えられている。例えば、プロセスまたはセグメント毎に暗号鍵を設定する

情報処理装置が特開平4-102920号公報「情報処理装置」に示されている。この例は、図32の例と同様にデータを暗号化しメモリ等に格納しておきそれを中央処理装置で実行する時に復号するのに加えて、プロセスまたはセグメント毎に変換（暗号復号）回路を選択する選択回路を有してゐる。選択回路は一般の暗号化されていないプログラムやユーザが作成したプログラムを実行する際に、中央処理装置が変換回路を介さずにプログラムを実行するように構成されている。これにより、オペレーションシステムが管理する論理単位であるプロセスまたはセグメント毎に異なる暗号鍵を使用することができ、暗号鍵の管理をオペレーションシステムで、フレキシブルに行うので暗号強度が高くなる。

【0013】また、近年の情報処理装置とくにマイクロプロセッサでは、キャッシュ・メモリの使用が広く行われている。そこで、上記の変換回路をキャッシュ・メモリと実メモリの間に配置して、復号されたプログラムやデータをキャッシュ・メモリ内に保持することにより、キャッシュ・メモリを効率よく使用し、暗号化／復号の処理を高速に行うことができる。

【0014】

【発明が解決しようとする課題】しかし、上記に示したプロセスまたはセグメント毎に暗号鍵を設定する情報処理装置では、選択回路の制御を論理単位であるプロセスまたはセグメント毎に行うために、選択回路の制御はプロセスまたはセグメントを管理しているオペレーション・システム（以下OSと呼ぶ）で行わなければならない。一般にOSはソフトウェアであるので、特に比較的簡単なアルゴリズムの暗号化方式を用いるこのようなシステムでは、OSの暗号切り替え制御部分を集中して解読することにより、容易に改ざん可能である。その結果、一旦装置内部で復号したデータをそのまま外部に出力するような改ざんが可能であり、OSの一部を改ざんしただけで、すべてのプログラムやデータが解読してしまうという問題点がある。つまり、ソフトウェアによる暗号化／復号手段の制御は装置におけるセキュリティ上の弱点になる。

【0015】また、OSで選択回路を制御し、かつ処理の高速化を図るために、復号されたデータをキャッシュ・メモリ内に保持しようとした場合、次のような問題点を伴う。

【0016】キャッシュ・メモリ制御方法には大きく分けて、キャッシュ・メモリ内のデータを書き換えるタイミングと実メモリのデータを書き換えるタイミングが一致しているライト・スルー方式と、タイミングが一致していないライト・バック方式がある。一般に高性能なのはライト・バック方式であるため、情報処理装置の性能を落とさずに暗号化／復号を行うには、ライト・バック方式のキャッシュ・メモリ制御方式が望ましい。

【0017】ところが、ライト・バック方式のキャッシ

ュを使用した場合、キャッシュ・メモリ内のデータを書き換えるタイミングと実メモリのデータを書き換えるタイミングが一致していない。そのため、ソフトウェアではキャッシュ・メモリ内に対するデータの書き込みのタイミングしか制御できない。その結果、ソフトウェアによる選択回路の制御は不可能となってしまう。つまり、OSによって選択回路を制御すると、復号されたデータをライト・バック方式のキャッシュに保持させることができない。従って、処理機能を低下させずに暗号強度を高くするには、キャッシュ・メモリ方式や仮想記憶方式のような、様々なメモリの管理方法に対応した暗号／復号鍵の管理方法が必要である。

【0018】さらに、従来の技術は、あらかじめ決まった暗号化方式または暗号鍵で暗号化されたプログラムやデータをソフトウェアの保護機能付き情報処理装置内であらかじめ決まった復号方式または復号鍵で復号して利用し、かつ復号する際のオーバーヘッドを考慮して計算量が少ない比較的簡単な暗号化方式を用いなければならない。そのため、暗号化／復号方式または暗号／復号鍵は非公開または装置固有のものとなる。従って、装置の暗号化／復号方式または暗号／復号鍵を知りえない第三者が、装置のセキュリティ機能に守られるソフトウェアを自由に開発できないという問題点がある。

【0019】本発明はこのような点に鑑みてなされたものであり、システムのメモリの管理方法に依存せずに高い暗号強度を得ることができるソフトウェアの保護機能付き情報処理装置を提供することを目的とする。

【0020】

【課題を解決するための手段】図1は、本発明の原理構成を示す図である。本発明では上記課題を解決するために、暗号化対象データの属性に応じて暗号鍵1aを生成する暗号鍵生成手段1と、暗号データの属性に応じて復号鍵2aを生成する復号鍵生成手段2と、前記暗号鍵生成手段1が生成した暗号鍵1aを用いて、暗号化対象データを暗号化する暗号化手段3と、前記暗号化手段3が暗号化した暗号データ4aを格納する記憶手段4と、前記記憶手段4に記憶されている暗号データ4aを、前記復号鍵生成手段2が生成した復号鍵2aを用いて復号する復号手段5と、前記復号手段5が復号したデータに含まれる命令に従って、復号したデータを処理する処理手段6と、前記処理手段6の処理終了後のデータを、暗号化対象データとして前記暗号化手段3へ出力する制御手段7と、を有することを特徴とするソフトウェアの保護機能付き情報処理装置が提供される。

【0021】このような構成によれば、暗号化対象データが入力されると、暗号鍵生成手段1が、その暗号化対象データの属性に応じて暗号鍵1aを生成する。この暗号鍵1aを用いて、暗号化手段3が暗号化対象データを暗号化し、暗号データ4aが記憶手段4に格納される。そして、暗号データ4aの処理要求があると、復号鍵生

成手段2が暗号データ4aの属性に応じて復号鍵2aを生成する。復号手段5は、その復号鍵2aを用いて、暗号データ4aを復号する。復号されたデータは、処理手段6により処理される。制御手段7は、処理手段6により処理されたデータを暗号化対象データとして暗号化手段3に対して出力する。

【0022】これにより、暗号鍵及び復号鍵が、暗号化／復号の対象となるデータの属性によって生成されるため、属性毎に異なる鍵を使用し暗号強度を高めることができるとともに、システムのメモリ管理方法に依存しないソフトウェアの保護機能付き情報処理装置が得られる。

【0023】また、暗号化又は復号の際に使用される複数の鍵を供給する鍵供給手段と、暗号化又は復号の際のアルゴリズムを表現した複数のアルゴリズム情報を供給するアルゴリズム供給手段と、命令とデータを含む暗号化対象データの暗号化の際には、供給されている前記複数の鍵と前記複数のアルゴリズム情報の中から暗号鍵と暗号化アルゴリズム情報とを選択し、暗号データの復号の際には、供給されている前記複数の鍵と前記複数のアルゴリズムの中から復号鍵と復号アルゴリズム情報とを選択する暗号化／復号方式選択手段と、前記暗号化／復号方式選択手段が選択した暗号鍵と暗号化アルゴリズム情報とを用い、暗号化対象データを暗号化する暗号化手段と、前記暗号化手段が暗号化した暗号データを格納する記憶手段と、前記暗号化／復号方式選択手段が選択した復号鍵と復号アルゴリズム情報とを用い、前記記憶手段に記憶されている暗号データを復号する復号手段と、前記復号手段に直接接続されており、前記復号手段が復号したデータに含まれる命令に従って、復号したデータを処理する処理手段と、前記処理手段の処理終了後のデータを直接に、暗号化対象データとして前記暗号化手段へ出力する制御手段と、を有することを特徴とするソフトウェアの保護機能付き情報処理装置が提供される。

【0024】このようなソフトウェアの保護機能付き情報処理装置によれば、暗号化対象データを暗号化する際には、暗号化／復号方式選択手段が、供給されている複数の鍵と複数のアルゴリズム情報の中から暗号鍵と暗号化アルゴリズム情報とを選択する。そして、暗号化手段により、暗号化対象データが暗号化される。暗号化された暗号データは、記憶手段に格納される。この記憶手段に格納された暗号データを復号する際には、暗号化／復号方式選択手段が、供給されている複数の鍵と複数のアルゴリズムの中から復号鍵と復号アルゴリズム情報とを選択する。そして、復号手段により、記憶手段に記憶されている暗号データが復号される。次いで、処理手段により、復号されたデータに含まれる命令に従ってデータを処理が行われ、制御手段により、処理後のデータが暗号化対象データとして暗号化手段へ出力される。

【0025】これにより、暗号化と復号の際に、そのデ

ータの属性等に応じて鍵とアルゴリズムとを変えることができるため、鍵のみを変える場合に比べ暗号強度をより高めることができる。

【0026】

【発明の実施の形態】以下、本発明の実施の形態を図面に基いて説明する。図1は本発明の原理構成を示す図である。本発明のソフトウェアの保護機能付き情報処理装置は、暗号化対象データ（命令とデータとを含む）の属性に応じて暗号鍵1aを生成する暗号鍵生成手段1と、暗号データの属性に応じて復号鍵2aを生成する復号鍵生成手段2と、暗号鍵生成手段1が生成した暗号鍵1aを用いて、暗号化対象データを暗号化する暗号化手段3と、暗号化手段3が暗号化した暗号データ4aを格納する記憶手段4と、記憶手段4に記憶されている暗号データ4aを、復号鍵生成手段2が生成した復号鍵2aを用いて復号する復号手段5と、復号手段5が復号したデータに含まれる命令に従って、復号したデータを処理する処理手段6と、処理手段6の処理終了後のデータを、暗号化対象データとして暗号化手段3へ出力する制御手段7と、で構成されている。

【0027】このような構成によれば、暗号化対象データが入力されると、暗号鍵生成手段1が、その暗号化対象データの属性に応じて暗号鍵1aを生成する。この暗号鍵1aを用いて、暗号化手段3が暗号化対象データを暗号化し、暗号データ4aが記憶手段4に格納される。そして、制御手段7より、暗号データ4aの処理要求があると、復号鍵生成手段2が暗号データ4aの属性に応じて復号鍵2aを生成する。復号手段5は、その復号鍵2aを用いて、暗号データ4aを復号する。復号されたデータは、処理手段6により処理される。制御手段7は、処理手段6により処理されたデータを暗号化対象データとして暗号化手段3に対して出力する。

【0028】これにより、暗号鍵と復号鍵とが暗号化対象データと暗号データとの属性によって定まるため、属性の異なるデータは異なる鍵で暗号化／復号され、暗号強度の高い暗号データを得ることができる。なお、上記の説明における属性とは、データのアドレス、アドレス領域、仮想アドレス、あるいはセグメントなどの、対象となるデータと何等かの関連性を有する各種情報を指す。

【0029】以下に本発明のソフトウェアの保護機能付き情報処理装置の具体的な構成について説明する。図2はソフトウェアの保護機能付き情報処理装置のハードウェア構成例を示すブロック図である。この情報処理装置は、まずメインの構成として、データの演算やメモリアクセス等の制御を行うMPU（Micro Processing Unit）である情報処理手段11、情報処理手段11が出力したデータを暗号化する暗号化手段12、情報処理手段11へ入力すべき暗号データを復号する復号手段13、暗号化／復号の際の暗号鍵と復号鍵とを選択する鍵選択

手段14、及び鍵選択手段14が選択した鍵を暗号化手段12と復号手段13とに供給する鍵供給手段15が設けられている。これらの要素は、安全保護容器10に収められている。また、暗号化手段12と復号手段13とは、システムバス20を介してRAM(Random Access Memory)である記憶手段21と、I/Oインタフェース(I/O)22に接続されている。

【0030】I/O22は、システムバス20とペリフェラルバス23との間のデータ伝送を制御する。ペリフェラルバス23には、イーサネットを介したデータ通信を行うイーサネット・コントローラ24、CD-ROMのデータを読み取るCD-ROMドライブ25、及び大容量のデータの磁気的なリード/ライトをするハードディスク26が接続されている。

【0031】以上の構成において、まず、予め決められた暗号化方式と暗号鍵で暗号化されたデータがネットワークやハードディスク26から供給される。暗号化されたデータを情報処理手段11で実行するためには、そのデータを記憶手段21にロードする必要がある。この時、暗号化されたデータの復号に使用する鍵の情報を、鍵選択手段14に送っておく。鍵選択手段14は鍵供給手段15に指示して復号手段13に復号鍵を渡すとともに、データの格納先アドレスに対応した暗号鍵を選択し、鍵供給手段15に指示して暗号化手段12に渡す。すなわち、暗号化されたプログラムまたはデータは復号手段13で一旦復号され、さらに暗号化手段12において格納先アドレスに対応した暗号鍵で暗号化され、記憶手段21にロードされる。プログラム実行時は、鍵選択手段14が鍵供給手段15にデータの格納先アドレスに対応した復号鍵を選択指示して復号手段13に渡すことで復号が行われる。

【0032】また、この情報処理装置の情報処理手段11、暗号化手段12、復号手段13、鍵選択手段14、及び鍵供給手段15は保護容器10内に格納されている。保護容器10は、内部の装置が外部からのプロービングに対して保護されるような仕組みを備えている。保護容器10はそれが取り外されたり穴を開けられたりした時に内部の装置の動作を停止させたり秘匿データが読み出されないように消去する機能を有しており、これにより暗号化・復号方式や暗号・復号鍵および復号されたデータの盗用や改ざんを完全に防ぐことができる。したがって、復号されたデータが存在する部分はすべて保護容器内に格納しなければならない。保護容器は攻撃抵抗容器(Tamper Resistant Module)とも呼ばれ、その例としては、米国特許4,593,384号や特開昭63-124153号公報、特開平2-44447号公報などがある。

【0033】このような安全保護容器は、複雑な経路をたどる導体路や論理素子が表面に形成された包囲体で秘密情報を取り扱う回路を囲み、内部の秘密情報を不正に

解析しようとして包囲体を破壊した場合に導体路の切断や短絡または論理素子の動作不良が生じ、それをトリガーとして暗号鍵などの秘密情報を消去するという構造になっている。したがって暗号鍵、復号鍵及び暗号化されていないデータ等を取り扱うMPU、暗号化装置及び復号装置を安全保護容器内に格納すれば、不正を行おうとする者は暗号を解読する手掛かりをほとんど失うことになる。暗号鍵、復号鍵及び暗号化されていないデータ等を取り扱うこれらの装置は1つのLSIに集積して実現するのが望ましいが、複数のLSIで構成した場合にはMCM(マルチ・チップ・モジュール)パッケージなどに封止して、外部にチップ間の配線が露出しないようにしてもよい。また、秘密情報の保持にはバッテリーバックアップされたSRAM(Static RAM)を用いて、容器が開けられると同時に電力の供給が断たれ、SRAMの内容が失われるような構造にしてもよい。

【0034】以上の例は、ネットワーク等を経由して入力された暗号データも、内部の暗号データと同じアルゴリズムで復号する場合の例である。ところが、この方式では、外部からこの情報処理装置に提供するソフトウェアも、内部の暗号データと同様に計算量の少ないアルゴリズムで暗号化しなければならない。そこで、複雑なアルゴリズムの復号を行う復号手段を別に設けることによって、この情報処理装置に提供するソフトウェアの暗号強度の強化を図ることができる。以下に、その例を示す。

【0035】図3は複数の復号手段を有するソフトウェアの保護機能付き情報処理装置のハードウェア構成を示すブロック図である。この情報処理装置には、MPUである情報処理手段31と、流通ソフト復号手段32と、暗号化手段33と、復号手段34と、RAMである記憶手段41が設けられている。さらに、I/Oインタフェース(I/O)42を介してハードディスク46と、CD-ROMドライブ45と、イーサネット・コントローラ44とが接続されている。また、情報処理手段31、流通ソフト復号手段32、暗号化手段33、及び復号手段34は、安全保護容器30に収められている。安全保護容器30内の各構成要素と、記憶装置41と、I/O42とはシステムバス40で接続されている。I/O42と、イーサネット・コントローラ44と、ハードディスク46と、CD-ROMドライブ45とは、ペリフェラルバス43で接続されている。

【0036】以上の構成において、データはまず、暗号化され、ネットワークやCD-ROM等によってソフトウェア提供者からユーザに提供される。この時用いられる暗号化方式は汎用的で暗号強度の高いものを用いる。例えば、ソフトウェア提供者は暗号化方式としてDES(Data Encryption Standard: 米商務省標準局〔現在の米国標準技術協会〕が1977年に公表した暗号アルゴリズム)等を用いて、ユーザに提供するデータを暗号化

する。そして復号鍵は例えばRSA (Rivest, Shamir, Adleman : Ronald Rivest, Adi Shamir, Leonard Adleman)の3氏が考案したアルゴリズム) 暗号によってソフトウェアの保護機能付き情報処理装置の公開鍵で暗号化し、暗号化されたデータとペアにしてユーザに提供する。ユーザは暗号化されたデータと装置の公開鍵で暗号化された復号鍵をI/O42から流通ソフト復号手段32に送って復号する。復号されたデータは直接暗号化手段33に送られ非公開の暗号方法または暗号鍵で暗号化される。そのため、ユーザは復号されたデータにアクセスすることはできない。暗号化手段33で暗号化されたデータは記憶手段41に格納される。情報処理手段31は、記憶手段41に格納されている暗号化されたデータを復号手段34で復号して受けとり、そのデータに含まれる命令を実行する。情報処理手段31が出力するデータの中で暗号化が必要なものは暗号化手段33で暗号化して記憶手段41に格納する。

【0037】このように公開された暗号化方式と暗号鍵を用いて暗号化されたデータをソフトウェアの保護機能付き情報処理装置内に取り込んで流通ソフト復号手段32で復号できるので、ソフトウェア提供者は暗号化手段33及び復号手段34で使用する暗号化/復号方式または暗号/復号鍵を知る必要はなく、ただ情報処理手段31用にソフトウェアを開発すればよい。つまり、ソフトウェアの保護機能付き情報処理装置内の暗号化/復号方式または暗号/復号鍵を知りえない第三者が、ソフトウェアの保護機能付き情報処理装置に守られるソフトウェアを自由に開発できる。

【0038】また、この情報処理装置も図2に示した例と同様に安全保護容器30で重要な構成要素が保護されているため、内部の装置が外部からのプロービングに対して保護されている。

【0039】ここで、図2に示した情報処理装置と図3に示した情報処理装置とを融合させることにより、さらにデータの保護を強化することができる。この例を図4に示す。

【0040】図4の情報処理装置は、情報処理手段51、流通ソフト復号手段52、暗号化手段53、復号手段54、鍵供給手段56、鍵選択手段55、記憶手段61、I/O62、ハードディスク66、CD-ROMドライブ65、イーサネット・コントローラ64、保護容器50、システムバス60、ペリフェラルバス63からなる。なお、この情報処理装置は、仮想記憶方式を採用しており、ページ単位でデータを取り扱うものとする。

【0041】このような構成において、暗号化されたデータと復号鍵は、ネットワークやCD-ROM等によってソフトウェア提供者からユーザに提供される。この暗号化されたデータを実行するためにロードするには、まず装置の公開鍵で暗号化された復号鍵をI/O62を介して流通ソフト復号手段52に送って復号する。次に、

復号されたデータは直接暗号化手段53に送られる。同時に鍵選択手段55は、データの格納先ページ番号に対応した暗号鍵群を選択する。選択された暗号鍵群は、鍵供給手段56により暗号化手段53に供給される。そして、暗号化手段53に送られたデータは、鍵供給手段56から供給された鍵群によって暗号化される。このときの暗号化方法または暗号鍵は非公開であるので、ユーザは復号されたデータにアクセスすることはできない。そして暗号化手段53でページ毎に異なる暗号鍵で暗号化されたデータは、記憶手段61に格納される。また記憶手段61に格納しきれないページはハードディスク66にスワップアウトされる。情報処理手段51が記憶手段61に格納されている暗号化されたプログラムを実行するには、鍵選択手段55がデータの格納先ページに対応した復号鍵群を鍵供給手段56から選択し、選択された復号鍵群で復号手段54がデータを復号して、復号されたプログラムを情報処理手段51が実行する。情報処理手段51が出力するデータの中で暗号化が必要なものは暗号化手段53で暗号化して記憶手段61に格納する。

【0042】図5は鍵選択手段で鍵群を選択する例の詳細を示す図である。鍵選択手段55はページ内のデータに対応する鍵を特定する鍵特定手段55aを有する。鍵供給手段56には、複数の鍵群(A群561、B群562、C群563)が格納されている。鍵群の内の各鍵は、ページオフセットと対応している。そして、情報処理手段51から暗号化/復号すべきページのページ番号が出力されると、鍵選択手段55がページ番号に応じて鍵群を指定する。この際、鍵特定手段55aが、ページ内の各ブロックに対応する鍵群内の鍵を特定する。この例では、鍵選択手段55はC群563を指定している。鍵供給手段56は、指定されたC群563の鍵群を、暗号化手段53と復号手段54とに供給する。暗号化手段53と復号手段54とは、暗号化あるいは復号すべきページの各ブロックを供給された鍵群の個々の鍵で暗号化あるいは復号する。

【0043】ここでページの大きさを256バイト、1つの鍵で暗号化する暗号ブロックを16バイトとすると、鍵群はそれぞれ16個までの鍵で構成される。いま1バイト毎にページ内のオフセットアドレスが与えられ、鍵群が16個の鍵からなるとすると、鍵特定手段はオフセットアドレスの任意の4ビットでどの鍵を使用するか特定する。また、暗号ブロック内の任意の4ビット(プログラムまたはデータそのものの任意の4ビットで、この4ビットは暗号化されない)でどの暗号/復号鍵を使用するか特定してもよい。

【0044】以上のようにして、ソフトウェアの保護機能付き情報処理装置内の暗号化/復号方式または暗号/復号鍵を知りえない第三者が、ソフトウェアの保護機能付き情報処理装置に守られるソフトウェアを自由に開発でき、暗号強度を高くすることができる。

【0045】ここで、所定タイミングで暗号データの鍵を変更することにより、さらに暗号強度を高くすることができる。以下にその例を示す。図6は再暗号化手段による鍵の変更方法の第1の例を示す図である。この例では暗号化と復号に同一の鍵を用いることとする。(A)は鍵の変更前の状態を示している。この例では、鍵供給手段56a内には「A」、「B」、「C」の3つの鍵56aa, 56ab, 56acが設けられている。そして、鍵選択手段55aは暗号鍵、復号鍵共に、「C」の鍵56acを選択している。従って、暗号化手段53aと復号手段54aとには、鍵「C」が供給されている。この状態で、鍵選択手段55aに対して、再暗号化手段57から鍵の変更指令が出力される。ここでは、ある期間ごとに鍵の変更指令を出力するものとする。なお、再暗号化手段57は、実際には、メインメモリにロードしたプログラムを情報処理手段が実行することにより実施される機能である。また、情報処理手段がプログラムに従ってDMAC(Direct Memory Access Controller)に命令して実行させてもよい。

【0046】(B)は鍵の変更指令出力後の状態を示す図である。ある期間が過ぎて鍵の変更指令が出力されると、再暗号化手段57の制御に従って、鍵選択手段55aは選択する暗号鍵を「C」から「A」に変更する。これにより、鍵「A」が、鍵供給手段56aから暗号化手段53aに渡される。この状態で記憶装置内の鍵「C」で暗号化されているデータを復号して、さらに鍵「A」で暗号化して記憶装置内に戻す。これにより、暗号鍵を鍵「C」から鍵「A」に変更しデータを再暗号化できる。

【0047】(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。再暗号化手段57は、鍵「C」で暗号化されているデータの全てを鍵「A」で暗号化し終わった時点で、復号鍵を鍵「A」にするように鍵選択手段55aに指令する。鍵選択手段55aが、指示に従って復号鍵を変更した結果、鍵供給手段56aから復号手段54aに鍵「A」が渡される。

【0048】このように再暗号化手段57によって鍵選択手段55は鍵をダイナミックに変更することができるので、暗号強度を高くすることができる。また、一度に用いる暗号 復号鍵または暗号/復号鍵群は、鍵供給手段56から供給される鍵または鍵群の1つに限る必要はなく、複数の鍵または複数の鍵群を一度に使用してもよい。図7は再暗号化手段による鍵の変更方法の第2の例を示す図である。この例でも暗号化と復号に同一の鍵を用いるものとする。(A)は鍵の変更前の状態を示している。鍵供給手段56bは「A」、「B」、「C」の3つの鍵56ba, 56bb, 56bcが供給可能である。鍵選択手段55bは、アドレス55ba、暗号鍵55bb、及び復号鍵55bcの項目が設けられた鍵管理

テーブルを有している。この対応表によって、アドレスと鍵との対応関係を管理している。この例では、アドレス「0」には、暗号鍵「A」、復号鍵「A」が対応しており、アドレス「1」には、暗号鍵「B」、復号鍵「B」が対応しており、アドレス「2」には、暗号鍵「C」、復号鍵「C」が対応している。そして、再暗号化手段57aは、鍵選択手段55bに対して、アドレス「2」のデータの鍵変更指令を出力するものとする。

(B)は鍵の変更指令出力後の状態を示す図である。一定時間が過ぎてアドレス「2」の鍵を「C」から「A」に変更する場合、再暗号化手段57aは鍵選択手段55b内の鍵管理テーブルのアドレス「2」の暗号鍵を「A」に変更する。従って、鍵選択手段55bは鍵供給手段56bに指示して暗号鍵「A」を暗号化手段に渡し、復号手段には引き続き復号鍵「C」を渡す。この状態で記憶装置内の復号鍵「C」で暗号化されているアドレス「2」のデータを復号して、さらに暗号鍵「A」で暗号化して記憶装置内に戻す。これにより、暗号鍵を「C」から「A」に変更し、アドレス2のデータを再暗号化できる。

【0049】(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。再暗号化手段57aは、再暗号化が終わった時点で、鍵選択手段55b内の鍵管理テーブルのアドレス「2」の復号鍵を「C」から「A」に変更する。この例でも、アドレスをアドレス領域に、鍵を鍵群にすることもできる。

【0050】上記の例では、鍵供給手段内には複数の鍵が予め登録されているが、この登録されている鍵を変更することもできる。以下にその例を示す。図8は鍵供給手段内の鍵を変更する場合の例を示す図である。この例では暗号化と復号に同一の鍵を用いることとする。鍵供給手段内の鍵を変更するには、図6に示したような再暗号化の手順が必要になる。

【0051】(A)は鍵の変更前の状態を示している。この例では、鍵供給手段56c内には「A」、「B」、「C」の3つの鍵56ca, 56cb, 56ccが設けられている。そして、鍵選択手段55cは暗号鍵、復号鍵共に、「C」の鍵56ccを選択している。従って、暗号化手段53cと復号手段54cとには、鍵「C」が供給されている。この状態で、鍵選択手段55cに対して、再暗号化手段57bから鍵の変更指令が出力される。ここでは、ある期間ごとに鍵の変更指令を出力するものとする。

【0052】(B)は鍵の変更指令出力後の状態を示す図である。ある期間が過ぎて鍵の変更指令が出力されると、再暗号化手段57bの制御に従って、鍵選択手段55cは選択する暗号鍵を「C」から「A」に変更する。鍵「A」が、鍵供給手段56cから暗号化手段53cに渡される。この状態で記憶装置内の鍵「C」で暗号化されているデータを復号して、さらに鍵「A」で暗号化し

て記憶装置内に戻す。

【0053】(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。再暗号化手段57bは、鍵「C」で暗号化されているデータの全てを鍵「A」で暗号化し終わった時点で、復号鍵を鍵「A」にするように鍵選択手段55cに指令する。鍵選択手段55cが、指示に従って復号鍵を変更した結果、鍵供給手段56cから復号手段54cに鍵「A」が渡される。そして、「C」の鍵56ccを、「D」の鍵56cdに変更する。このようにして、鍵供給手段56c内の鍵「C」が廃棄され鍵「D」の供給が開始される。

【0054】ここで鍵Dの生成手段としては、新たに鍵を作成する方法と多数の鍵が登録された鍵のテーブルから選択する方法とがある。図9は、鍵の生成方式を示す図である。ここで、以下に示す2つの例は、鍵供給手段56cに関しては、図8に示したものと変わるところがないため、同じ符号を付して説明を省略する。

【0055】(A)は新たな鍵を作成する方式を示す図である。鍵生成手段58aは、内部に原型となる鍵58aaを有しており、この鍵58aaを変形させ新たな「D」の鍵56abを生成する。鍵生成手段58aとしては、シリアル・シフト・レジスタを用いた疑似乱数発生装置や、装置固有の係数と時刻を変数とした関数を用いることができる。(B)は鍵テーブルを利用する方式を示す図である。鍵生成手段58b内には、多数の鍵58ba～58beが登録されている。そして、この中の任意の鍵58bdを選択し、「D」の鍵とする。

【0056】このように2通りの方法が考えられるが、もちろん鍵生成手段と鍵テーブルを組み合わせてもよい。また、鍵選択手段からの鍵の指定を、名前ではなく識別子で行うことにより、鍵供給手段内の鍵が変更されたことを鍵選択手段が認識する必要がなくなる。

【0057】図10は識別子を用いた鍵の選択方式を示す図である。(A)は鍵の変更前の状態を示す図である。鍵供給手段56dは「A」、「B」、「C」の3つの鍵56da、56db、56dcが登録されており、各鍵56da、56db、56dcには、0～2の識別子が付けられている。鍵選択手段55dは、暗号鍵識別子を「2」、復号鍵識別子を「2」として出力している。従って、暗号化手段53dと復号手段54dとには、鍵「C」が供給されている。この状態で、鍵「C」が鍵「D」に変更される。

【0058】(B)は鍵の変更後の状態を示す図である。鍵選択手段55dは、同じように暗号鍵識別子を「2」、復号鍵識別子を「2」として出力しているが、鍵供給手段56d内の「2」の識別子には、鍵「D」が登録されている。従って、暗号化手段53dと復号手段54dとには、鍵「D」が供給されている。このようにして、鍵選択手段55dは、鍵Cが鍵Dに置き換わってしまった事を認識する必要がなくなる。

【0059】次に、鍵を書き換えるタイミングを時間管理手段で管理し、再暗号化手段で異なる鍵でデータを再暗号化して鍵を更新する例を示す。時間管理手段は、時間管理テーブルを用いて、再暗号化のタイミングを管理することができる。図11は時間管理テーブルを示す図である。時間管理テーブル59は、鍵識別子59a、鍵寿命59b、経過時間59c、書き換えフラグ59dの項目を有している。そして、鍵識別子59aの「0～2」は、それぞれ鍵「A」～「C」と対応している。鍵寿命59bには、その鍵を書き換える周期が設定されている。経過時間59cは、現在の鍵の使用開始からの経過時間を示している。書き換えフラグ59dは、値が「0」で、経過時間が鍵寿命を超えていないことを示し、値が「1」で、経過時間が鍵寿命を超えたことを示す。

【0060】この時間管理テーブル59を用いて、時間管理手段はそれぞれの鍵の寿命と、鍵を書き換えてからの経過時間を管理する。ここで経過時間が鍵寿命を超えた鍵の有効鍵通知手段である書き換えフラグを立てて、鍵の書き換えおよび使用している鍵の変更が必要な事を再暗号化手段に知らせる。

【0061】図では、識別子「0」の寿命は「10分」、経過時間は「1分」、書き換えフラグは「0」である。識別子「1」の寿命は「60分」、経過時間は「30分」、書き換えフラグは「0」である。識別子「2」の寿命は「60分」、経過時間は「61分」、書き換えフラグは「1」である。つまり、識別子「2」は、経過時間が寿命を超えている。

【0062】図12は時間管理手段の処理手順を示すフローチャートである。

〔S1〕寿命となった鍵識別子があるか否かを判断し、寿命となった鍵識別子があればステップS2に進み、無ければこのステップを繰り返す。

〔S2〕寿命となった鍵識別子の書き換えフラグを立てる。

〔S3〕フラグが立っている鍵識別子の鍵が使用されているか否かを判断し、使用されていればこのステップを繰り返し、使用されていなければステップS4に進む。なお、フラグが立っている鍵識別子の鍵が使用されているか否かの情報は、再暗号化手段から与えられる。

〔S4〕寿命となった鍵識別子の鍵の書き換えを行い、ステップS1に進む。

【0063】図13は再暗号化手段の処理手順を示すフローチャートである。

〔S5〕書き換えフラグを読む。

〔S6〕書き換えフラグが立っているか否かを判断し、書き換えフラグが立っていればステップS7に進み、書き換えフラグが立っていなければステップS5へ進む。

〔S7〕フラグが立っている鍵識別子の鍵で暗号化されているデータを、フラグが立っていない鍵識別子の鍵で

暗号化し直す。

〔S8〕フラグが立っている鍵識別子を使用していないことを、時間管理手段に知らせ、ステップS5に進む。

【0064】図12、図13に示したように時間管理手段と再暗号化手段が動作することにより、まず、寿命となった鍵があったなら、時間管理手段がその鍵の鍵アドレスに書き換えフラグを立てる。すると、再暗号化手段は、書き換えフラグが立っていることを検出し、フラグが立っている鍵識別子の鍵をフラグが立っていない鍵識別子の鍵に鍵変更をして、時間管理手段に知らせる。時間管理手段はフラグが立っている鍵アドレスの鍵が使用されていない事を確認して、寿命となった鍵アドレスの鍵を書き換えることを鍵供給手段に要求する。このようにして安全に鍵の書き換えを行うことができる。

【0065】図14は、時間管理手段の鍵書き換えに関する他の動作例を示すフローチャートである。この例では、書き換えフラグを立てて10分の間に必ず鍵の書き換えを行う。

〔S11〕寿命となった鍵識別子があるか否かを判断し、寿命となった鍵識別子があればステップS12へ進み、なければこのステップを繰り返す。

〔S12〕寿命となった鍵識別子の書き換えフラグを立てる。

〔S13〕フラグが立っている鍵識別子の鍵が使用されているか否かを判断し、使用されていればステップS14に進み、使用されていなければステップS15に進む。

〔S14〕フラグが立てられてから10分が経過したか否かを判断し、経過していればステップS15に進み、経過していなければステップS13に進む。

〔S15〕寿命となった鍵識別子の鍵の書き換えを要求し、ステップS11に進む。

【0066】このようにすることで、再暗号化手段が改ざんされて、鍵の変更を行わなくなってしまう場合でも、鍵は強制的に書き換えられ、その結果情報処理手段が正常動作しなくなることで、自己破壊的にデータの秘匿性を保つことができる。この例では10分の間に必ず鍵の書き換えを行うようにしたが、実際は、システムの使用状況や情報処理手段1の処理能力を勘案して決定するとよい。また、鍵を強制的に書き換える代わりにNMI (Non Maskable Interrupt) を発生させて、システムをリセットしてもよい。

【0067】次に、図4の例に示す鍵選択手段55の鍵管理方式の幾つかの例を説明する。どのデータをどの暗号化方式及びどの鍵で暗号化したかを管理する方法としては、記憶手段61のアドレスを用いるのが簡単である。特に仮想記憶方式を使わないシステムでは、記憶手段61のアドレスを直接用いるか、あるいは何等かの単位毎に管理すればよい。例えばアドレスと暗号鍵の対応を管理する鍵管理テーブルを用いるとよい。

【0068】仮想記憶方式で記憶を管理するシステムでは、記憶手段61に格納しきれないページまたはセグメントは記憶手段61からハードディスク66に移される。ハードディスク66に移されたページまたはセグメントは記憶手段61の同じアドレスに戻されるとは限らないので、記憶手段61のアドレスを直接用いて鍵の管理を行うことはできない。そこで仮想アドレスによる鍵の管理が必要になる。仮想アドレスによる鍵の管理の方法としては、仮想アドレスを直接用いる方法と、ハードディスク66と記憶装置61の間でやり取りされる物理単位で固定長のページを用いる方法と、ハードディスク66と記憶装置61の間でやり取りされる論理単位で可変長のセグメントを用いる方法と、仮想記憶空間を複数持つような多重仮想記憶方式をとる場合は仮想記憶空間毎に鍵を管理する方法がある。

【0069】ページによって仮想記憶を管理する場合、一般に仮想アドレスはページ番号とページオフセットから成っているので、例えばページ番号と鍵識別子の対応を管理する鍵管理テーブルを用いるとよい。図15はページ番号と鍵識別子の対応を管理する鍵管理テーブルの例を示す図である。通常、仮想アドレス70はページ番号71とページ・オフセット72で構成されている。そこで、鍵管理テーブル80には、ページ番号81に対応して鍵識別子82を登録しておく。従って、ある仮想アドレスへのアクセスが発生すると、その仮想アドレスのページ番号を抽出する。そして、鍵管理テーブル80から、抽出したページ番号に対応する鍵識別子を選択することができる。

【0070】また、鍵管理テーブルが大きくなる場合には、鍵管理テーブルを記憶装置61に置き、頻繁に使用する鍵管理テーブルの一部を鍵選択手段内のTLB (Table Lookaside Buffer) に保持して使用し、TLBにないものは記憶装置61から取り出して、TLBを更新するようにしてもよい。図16は鍵選択手段内のTLBを示す図である。鍵選択手段内のTLB90は、アドレス・タグ91と鍵識別子92との格納部で構成されている。アドレス・タグ91には、仮想アドレス70のページ番号71が格納され、格納されたページ番号71に対応する鍵識別子が鍵識別子92の格納部に格納される。これにより、鍵管理テーブルを保持するためのメモリサイズを縮小することができる。

【0071】しかし、一般にページによって仮想記憶を管理するシステムの場合、仮想記憶を管理するためのページテーブルを持ち、仮想アドレスから物理アドレスへの変換と、各ページが主記憶内に存在するか否かを管理し、またそのためのハードウェアであるMMU (Memory Management Unit) を有している。そこで、鍵識別子の情報をページテーブルに追加することで、MMUと鍵管理テーブルの機能を融合させることができる。図17はMMU内のTLBを示す図である。MMU内のTLB1

00には、アドレス・タグ101、主記憶内に存在するか否かを示すビット102、鍵識別子103、及びページ・アドレス104の格納部で構成されている。アドレス・タグ101には仮想アドレス70のページ番号71が格納されており、主記憶内に存在するか否かを示すビット102により、そのページがメモリ内にあるか否かが示されている。そして、鍵識別子103により、そのページの鍵が示されており、ページ・アドレス104により、そのページのアドレスが示されている。また、セグメントによって管理する場合もページと同様に鍵の管理ができる。

【0072】仮想記憶とライトバック方式のキャッシュ・メモリを使用して、アドレスによる暗号鍵管理を行った場合ソフトウェアによる鍵の選択は不可能である。そこで、図17のようにMMUを利用することで、ハードウェアによる鍵の選択が可能になる。つまり、MMU内ではページ番号およびセグメント番号に対応したアドレスタグをアドレスとした連想記憶方式でページテーブルおよびセグメントテーブルの一部を保持しており、通常キャッシュ・メモリ内に存在するデータが属するページまたはセグメントは、MMU内に存在する。ページテーブルまたはセグメントテーブルがMMU内に存在するので、MMUから鍵アドレスを取得することで、ハードウェアによる鍵の選択が可能になる。

【0073】図18はインテル社製のMPUであるi80286のタスク状態セグメントに鍵アドレスの情報を付加した例を示す図である。タスク状態セグメント110は、バック・リンク111、特権レベル0、1、2に対するスタック・ポインタSP、SSの初期値112、IP113、SR114、汎用レジスタ115、ES116、SS117、CS118、DS119、タスクLDTセクタ120、及び鍵識別子121の格納領域がある。このタスク状態セグメント110はタスク毎に生成されるため、このようにする事で、タスク（プロセス）毎に暗号鍵を管理する事ができる。またi80286ではタスク毎に仮想記憶空間を有する多重仮想記憶方式を採用しているので、仮想記憶空間毎の鍵管理の例でもある。

【0074】暗号化されていないデータを図4に示したような本発明の装置で実行する方法は2種類ある。1つは暗号化されていないデータも暗号化手段53で暗号化して、暗号化されて供給されたデータと同様に取り扱う方法である。2つ目は暗号化されて供給されたデータが必ず流通ソフト復号手段52で復号される場合、このことを利用して、流通ソフト復号手段52で復号されたデータのみに関連鍵アドレスを与えるようにする方法である。このようにして、暗号化されたデータの秘匿性を保ったままで暗号化されていないデータも扱うようにすることができる。

【0075】以上の例は、暗号化 復号の対象となるデ

ータに関連性のあるアドレス等の情報を用いて鍵管理をするものであるが、データに関連性のある情報ではなく、データの内容による鍵管理も可能である。例えば、プログラムおよびデータの内容の一部分で暗号鍵を選ん で、残りの部分を暗号化する方法が考えられる。以下に、その例を説明する。

【0076】図19はデータの内容による鍵管理の暗号化の処理手順を示すフローチャートである。これは、暗号化手段が行う処理である。

【S21】暗号化されるべきデータを暗号化手段へ読み込む。

【S22】読み込んだデータから特定の数ビットを抜き出す。

【S23】ステップS22において抜き出した数ビットに対応する暗号鍵を、鍵テーブルより選択する。

【S24】ステップS23において選択した暗号鍵を用いて、暗号化されるべきデータの、抜き出した数ビット以外の部分を暗号化する。

【S25】暗号化したデータに、ステップS22において抜き出した数ビットを埋め込む。

【0077】以上の各ステップにおいて、暗号化されるデータの状態がどのように変化するかを、図20の状態遷移図に従って説明する。(A)は、ステップS21の状態を示す図である。暗号化すべきデータ130が読み込まれる。暗号化される前のデータ130は32ビットからなり、情報処理手段で直接処理を行うことのできる状態にある。

【0078】(B)は、ステップS22の状態を示す図である。暗号化される前のデータ130から、ある規定に従った特定の数ビットを抜き出す。例では7、11、15及び23ビット目の4ビットのデータを抜き出している。従って、28ビットのデータ131が残ることとなる。抜き出したビットは、所定の配列に並べられビット情報132となる。

【0079】(C)は、ステップS23の状態を示す図である。4ビットのビット情報132に対応する暗号鍵を、予め用意された鍵テーブル140から選択する。鍵テーブル140は、抜き出したビットによるビット情報140aに対応する鍵140bが一意に選択できるようになっている。例では抜き出すビット数が4ビットであることから鍵も4ビットで分類されており、16種類のビット情報「0000」、「0001」、「0010」・・・のそれぞれに対応する16個の鍵141、142、143・・・が存在する。この例では、抜き出したビットによるビット情報132の値が「0010」であるため、鍵143が暗号鍵として選択されている。

【0080】(D)は、ステップS24の状態を示す図である。選択された鍵143を暗号鍵に用いて、抜き出した4ビットのビット情報132以外の28ビットのデータ131を暗号化し、28ビットの暗号データ133

を生成する。

【0081】(E)は、ステップS25の状態を示す図である。(B)において抜き出した4ビットのビット情報132を、28ビットの暗号データ133に埋め込み、32ビットの暗号データ134として記憶手段に格納する。

【0082】抜き出した4ビットのビット情報132を埋め込む方法は、復号の際にビット情報132を分離するための方法とセットで決定されている。この2つの方法が一意に決定できるのであれば、埋め込む位置を最初に抜き出したビット位置と同じにする必要はない。また、最後にスクランブル等を行い、第三者に暗号鍵の選択に用いたビットの位置を容易に見つけられないようにすることも可能である。なお、最後にスクランブルを行う場合には、ステップ24で行う暗号化はスクランブル以外の方法である必要がある。更に、抜き出した4ビットのビット情報132を暗号化した28ビットの暗号データ133に埋め込む際、XOR等の暗号化方法で暗号化して埋めこんでもよい。

【0083】図21は、データの内容による鍵管理の復号の処理手順を示すフローチャートである。

【S31】暗号化されているデータを復号手段へ読み込む。

【S32】復号手段において、読み込んだデータから特定の数ビットを抜き出す。

【S33】ステップS32において抜き出した数ビットに対応する復号鍵を、鍵テーブルより選択する。

【S34】ステップS33において選択した復号鍵を用いて、復号されるべきデータの、抜き出した数ビット以外の部分を復号する。

【S35】復号したデータに、ステップ32において抜き出した数ビットを埋め込む。

【0084】復号されるデータの状態がどのようにに変化するかを、図22の状態遷移図に従って説明する。32ビットのデータを復号する場合を考える。但しこの32ビットは、図19及び図20に示した暗号化方法に従って暗号化された後、記憶手段に格納されていたものとする。なお、この例は暗号鍵と同じ鍵を復号鍵とする場合である。

【0085】(A)は、ステップS31の状態を示す図である。暗号データ134を記憶装置から読み込む。暗号データ134は32ビットからなり、情報処理手段で直接処理を行うことのできない状態にある。

【0086】(B)は、ステップS32の状態を示す図である。暗号データ134から、特定の数ビットを抜き出し、ビット情報132を得る。例では6、13、27及び30ビット目の4ビットを抜き出している。この4ビットのデータを抜き出す位置は、暗号化の際に、ビット情報132の各ビットを埋め込んだ位置である。この結果、28ビットの暗号データ133が残ることとな

る。

【0087】(C)は、ステップS33の状態を示す図である。抜き出した4ビットのビット情報132に対応する鍵143を、予め用意された鍵テーブル140から選択する。暗号化の際に用いられたビット情報の値が「0010」であるため、鍵テーブル140を用いて、暗号化の際の鍵143と同一の鍵を選択することができる。この選択された鍵143が復号鍵となる。

【0088】(D)は、ステップS34の状態を示す図である。選択した鍵143を復号鍵として、抜き出した4ビットのデータ132以外の28ビットの暗号データ133を復号し、28ビットのデータ131を生成する。

【0089】(E)は、ステップS35の状態を示す図である。(B)において抜き出した4ビットのデータ132を、28ビットのデータ131に埋め込み、32ビットのデータ130を得る。埋め込む位置は、暗号化の際に4ビットのデータを抜き出した位置と同じ位置である。この結果、暗号化する前のデータと同一のデータ130が得られる。

【0090】以上のような方法により、復号鍵の抽出に用いるデータが復号の対象となる暗号化されたデータの一部で構成される。そのため、余分な情報をデータに付加する必要がなく、メモリを有効に使用できる。上記の例では具体的なビット数を示して説明を行ったが、これは説明を解りやすくするためであり、ビット数はソフトウェアの保護機能付き情報処理装置の各々で任意に選ぶことができる。従ってある単位、例えばバスで取り扱うビット単位(8ビット、16ビット、32ビット、64ビット、128ビット等)や仮想記憶で用いられるページ単位、キャッシュメモリのデータを交換するビット単位等に関連させて暗号化のブロックを選択することで効率的な暗号化及び復号を行うことができる。

【0091】上記の例では、暗号化の鍵と復号の鍵とを選択する際に、同じ鍵テーブルを用いたため、暗号鍵と復号鍵が同一となっているが、暗号化の鍵テーブルと復号の鍵テーブルとを個別に設けることにより、暗号鍵と復号鍵とを個別のものにすることができる。これにより、暗号化の際に用いた暗号鍵と異なる復号鍵により復号するような暗号化／復号アルゴリズムを利用することもできる。

【0092】また、上記の例では抜き出した数ビットから暗号鍵を選択する時に鍵テーブルを用いたが、抜き出した数ビットをある特定の関数に入力し、その出力値を暗号鍵として用いてもよく、そのようにすれば鍵テーブルを持つ必要がなくなる。勿論抜き出した特定の数ビットそのものを暗号鍵としてもよい。

【0093】以上のような方法により、暗号鍵及び復号鍵の選択がデータの上に依存しており、保護の対象であるデータの一部あるいは全部から暗号鍵を導出すること

ができるため、特にアドレスに依存せずに、複数の暗号鍵でデータを暗号化することができる。その為、OSによる複雑なアドレス管理が行われている場合でも、OSに依存することなく複数の暗号鍵を管理することができる。つまり、OSを改ざんすることにより秘匿データを盗用するという攻撃に対して、有効な防御を実現することができる。

【0094】また、暗号化に複数の暗号鍵を使用しているので、計算量の少ない暗号化方法（例えばXORなど）を用いても十分な暗号強度を得ることができ、暗号化及び復号のプロセスがデータの処理速度に及ぼす影響を少なくすることができる。従って、メモリ中のデータを逐次復号しながらプログラムを実行する場合でも十分な実行速度と暗号強度を実現することができる。これにより、常にメインメモリ中に常駐するようなOS自身を防御することも可能である。

【0095】なお、上記で説明した本発明の各構成要素は、ハードウェアの回路で構成することができる。その場合、暗号鍵生成手段は暗号鍵生成回路であり、復号鍵生成手段は復号鍵生成回路であり、暗号化手段は暗号化回路であり、記憶手段はメモリであり、復号手段は復号回路であり、処理手段はALU (Arithmetic and Logic Unit) であり、制御手段はMPU (Micro Processing Unit) である。

【0096】ところで、以上の例は、データの属性ごとに異なる暗号／復号鍵を使用することにより暗号の強度を高めているが、プログラムの実行中に鍵と暗号化／復号アルゴリズムとを随時変更することにより、暗号強度を高めることもできる。このような例を以下に示す。

【0097】図23は暗号化／復号アルゴリズムを随時変更するソフトウェア保護機能付き情報処理装置のハードウェア構成例を示すブロック図である。この情報処理装置は、図4に示したソフトウェア保護機能付き情報処理装置と同様の構成として、情報処理手段151、流通ソフト復号手段152、暗号化手段153、復号手段154、及び鍵供給手段155を有している。但し、この例における暗号化手段153と復号手段154とは、暗号化又は復号のアルゴリズムを切り換えることができる点で、図4に示した暗号化手段53と復号手段54と異なる。

【0098】この例では、さらに、暗号化／復号アルゴリズムのプログラムを供給するアルゴリズム供給手段156が設けられている。そして、図4で示した鍵選択手段55に代わり、暗号化手段153と復号手段154とに供給すべき鍵とアルゴリズムとを選択する暗号化方式選択手段157が設けられている。

【0099】システムバス160、記憶手段161、I/Oインタフェース(I/O)162、ペリフェラルバス163、イーサネット・コントローラ164、CD-ROMドライブ165、及びハードディスク166の構

成は、図4の例と同じである。

【0100】以上の構成において、まず、予め決められた暗号化方式で暗号化されたデータがネットワークやCD-ROM等によってソフトウェア提供者から提供される。この時の暗号化方式には、汎用的で暗号強度の高いものを用いる。例えば、DES等を用いる。一方、このデータを暗号化した際の暗号鍵（この例では、同一の鍵で暗号化・復号が行われる）は、RSAのような暗号によってこの情報処理装置の公開鍵で暗号化する。従って、ユーザには、暗号強度の高い暗号化方式で暗号化されたデータと、公開鍵で暗号化された暗号鍵とのペアが提供される。

【0101】ユーザは、暗号化されたデータと公開鍵で暗号化された暗号鍵とを、I/O162を解して流通ソフト復号手段152に送る。流通ソフト復号手段152は、先ず、公開鍵で暗号化された暗号鍵を、自己の持っている秘密鍵で復号する。そして、復号された暗号鍵を用いてデータを復号する。復号されたデータは、直接暗号化手段153に送られる。暗号化手段153は、暗号化方式選択手段157が選択した暗号化アルゴリズム及び暗号鍵により、データを暗号化する。暗号化されたデータは記憶手段161に格納される。記憶手段161に格納されたデータは、非公開の暗号化アルゴリズム及び暗号鍵で暗号化されているため、ユーザ自信も、この装置での実行以外の目的で記憶手段161内のデータにアクセスすることはできない。

【0102】記憶手段161に格納したデータ内の命令を実行する際には、先ず、復号手段154が暗号化方式選択手段157から供給された復号アルゴリズム及び復号鍵で、データを復号する。復号されたデータが情報処理手段151に送られ、情報処理手段151がそのデータを実行する。なお、情報処理手段151が実行した結果を記憶手段161に書き出す際には、そのデータは、暗号化手段153により暗号化された後、記憶手段161に格納される。この時、情報処理手段151が実行した後のデータを、別の暗号化方式で暗号化して記憶手段161に格納することができる。暗号化方式の変更処理は、暗号化方式選択手段157が行う。

【0103】以下に、暗号化方式選択手段157における暗号化方式の変更方法の一例を説明する。図24は暗号化方式の変更例を示す図である。図6では再暗号化手段によって再暗号化が行われたが、この例では、暗号化方式選択手段が同様の働きをすることとする。また、簡単にするために暗号化と復号に同一のアルゴリズムと鍵を用いるものとする。

【0104】(A)はアルゴリズムの変更前の状態を示す図である。この図において、鍵供給手段155は、「KA」、「KB」、「KC」の3つの鍵155a~155cを供給している。アルゴリズム供給手段156は、「A」、「B」、「C」の3つのアルゴリズム15

6a～156cを供給している。暗号化方式選択手段157は、暗号化あるいは復号を行う際に、鍵供給手段155とアルゴリズム供給手段156との中の鍵とアルゴリズムとから、その都度必要なものを選択し、暗号化手段153と復号手段154とに供給する。この図では、暗号化手段153と復号手段154との双方に、アルゴリズム「C」と鍵「KC」とが供給されている（以下、このような組み合わせの暗号化方式を、「暗号化方式CC」と呼ぶ）。

【0105】この状態において装置が動作している場合に、所定のタイミングで暗号化方式の変更が行われる。変更のタイミングとしては、例えば、その暗号化方式が使われだしてから一定の時間が経過した時などである。ここでは、アルゴリズムを「A」（暗号化方式AC）に変更するものとする。

【0106】（B）はアルゴリズムの変更指令出力後の状態を示す図である。変更指令が出力されると、暗号化方式選択手段157は、アルゴリズム「A」を暗号化手段153に渡す。復号手段154には、引き続きアルゴリズム「C」を渡す。この状態で、記憶手段161内に暗号化方式CCで暗号化されているプログラムを、復号手段154が復号する。復号されたプログラムは暗号化手段153に送られる。暗号化手段153は、暗号化方式ACにより、受け取ったプログラムを暗号化する。このような処理を、記憶手段161内の暗号化方式CCで暗号化されているプログラム全てについて実行する。再暗号化が終了したら、暗号化方式選択手段157は、復号手段154に供給するアルゴリズムを「A」に変更する。

【0107】（C）は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。この状態では、暗号化手段153と復号手段154とには、共に、アルゴリズム「A」と鍵「C」とが供給されている。以後は、暗号化方式ACにより暗号化されたプログラムの実行と、暗号化方式ACによる暗号化が可能となる。

【0108】これにより、暗号化方式選択手段157により暗号化方式と復号方式とをダイナミックに変更することができるため、暗号強度を高くすることができる。このようなアルゴリズムの変更を可能とするハードウェア構成としては、例えば、暗号化手段153及び復号手段154をCPUやプログラマブル・ロジックで構成し、アルゴリズム供給手段156は、これらの動作（アルゴリズム）を表現する情報であるプログラムで供給することができる。また、暗号化手段153と復号手段154とをFPGA(Field Programmable Gate Array)を用いて構成し、アルゴリズム供給手段は、これらの動作（アルゴリズム）を表現する情報としてFPGAの配線情報を供給することもできる。動作中にでも配線情報を変更することができるFPGAとしては、米国Atmel

1社のキャッシュ・ロジックがある。このキャッシュ・ロジックは、内部にプログラムを格納するための領域を複数有するSRAMを備えており、外部からの信号入力により、動作させる領域を切り換えることができる。そのため、高速に配線情報の切り換えが可能である。従って、頻繁に暗号方式の変更を行っても、システムの処理能力を低下させずにすむ。

【0109】また、アルゴリズムに対応させる鍵を、複数の鍵からなる鍵群とすることもできる。図25は暗号化方式を変更する際に鍵群を選択する例を示す図である。この例では、鍵供給手段1550は、「A群」、「B群」、「C群」の3つの鍵群1551～1553を有している。一方、アルゴリズム供給手段1560は、図23の例と同様に「A」、「B」、「B」の3つのアルゴリズム1561～1563を有している。暗号化方式選択手段157aは、これらの鍵とアルゴリズムとの中から任意のものを選択し、暗号化手段153aと復号手段154aとに供給している。この図では、暗号化手段153aと復号手段154aとの双方に、アルゴリズム「C」と「C群」の鍵群1553とが供給されている。

【0110】この場合、1つの鍵群は、ある特定のアドレス領域に適用され、このアドレス領域をさらに細分化したアドレス領域に、個々の鍵を適用する。次に、アルゴリズム供給手段1560が供給するアルゴリズムの内容を変更する場合について説明する。

【0111】図26は供給するアルゴリズムを変更する際の処理手順を示す図である。（A）はアルゴリズムの変更前の状態を示す図である。この状態は、図24の（A）の状態と同じである。この例では、「C」のアルゴリズム156cの使用を停止して、代わりに「D」のアルゴリズムを供給する場合について説明する。その場合、「C」のアルゴリズム156cの使用を停止する為に、「C」のアルゴリズム156cで暗号化されて記憶手段に格納されているプログラムを、他のアルゴリズムにより再暗号化しなければならない。そこで、まず、暗号化方式CCで暗号化されているプログラムを、暗号化方式ACで再暗号化する。

【0112】（B）はアルゴリズムの変更指令出力後の状態を示す図である。暗号化手段153には、「A」のアルゴリズムが供給されている。この状態は、図24の（B）と同じである。この状態で、暗号化方式CCで暗号化されているプログラムを、暗号化方式ACで暗号化し直して、記憶手段に格納する。これにより、「C」のアルゴリズムを別のアルゴリズムに変更することが可能となる。

【0113】（C）はアルゴリズムを変更後の状態を示す図である。「C」のアルゴリズムが格納されていた領域には、「D」のアルゴリズム156dが格納されている。以後、このアルゴリズム156dが暗号化手段15

3や復号手段154に供給される。

【0114】このようなアルゴリズムの変更は、アルゴリズム登録テーブルを用意しておくことにより容易に実現できる。また、各アルゴリズムに使用期限を設け、期限の超えたアルゴリズムを別のアルゴリズムに変更するように、制御することができる。

【0115】図27はアルゴリズムの使用時間の経過時にアルゴリズムの変更を行う例を示す図である。アルゴリズム供給手段154には、「A」、「B」、「C」の3つのアルゴリズムが供給されている。これらのアルゴリズムは、それぞれアルゴリズム・アドレス「0」～「2」に対応している。アルゴリズム登録テーブル158には、「A」～「Z」のアルゴリズムが登録されている。アルゴリズムを変更する際には、アルゴリズム登録テーブル158内のアルゴリズムが、アルゴリズム供給手段154に渡される。そして、アルゴリズムを変更するタイミングは、時間管理手段159が管理している。

【0116】時間管理手段159は、各アルゴリズムのアルゴリズム・アドレス「0」～「2」に対応して、「アルゴリズム使用期限」、「経過時間」、及び「書き換えフラグ」の情報を管理している。「アルゴリズム使用期限」には、そのアルゴリズムが使用できる制限時間が予め設定されている。「経過時間」には、そのアルゴリズムの使用が開始されてからの経過時間が示されている。「書き換えフラグ」は、経過時間が使用期限を超えたか否かを示すフラグである。通常の値は「0」であり、使用期限を超えるとフラグが立ち、値が「1」になる。

【0117】次に、暗号化方式選択手段によるアルゴリズムの変更とアルゴリズム供給手段によるアルゴリズムの書き換えの処理について説明する。図28はアルゴリズム供給手段のアルゴリズムの書き換え処理のフローチャートである。

〔S41〕使用期限を過ぎたアドレスがあるか否かを判断する。使用期限が過ぎたアドレスがあればステップS42に進み、使用期限が過ぎたアドレスがなければこのステップS41を繰り返す。

〔S42〕使用期限が過ぎたアドレスがあった場合には、そのアドレスの書き換えフラグを立てる。

〔S43〕フラグが立っているアドレスのアルゴリズムが使用されているか否かを判断する。使用されているか否かの情報は、暗号化方式選択手段157から提供される。

〔S44〕使用期限が過ぎたアドレスのアルゴリズムを書き換える。

【0118】図29は暗号化方式選択手段のアルゴリズム切り換え処理のフローチャートである。

〔S51〕アルゴリズムの書き換えフラグを読む。

〔S52〕書き換えフラグが立っているアルゴリズム・アドレスがあるか否かを判断する。書き換えフラグが立

っているアルゴリズムがあればステップS53に進み、書き換えフラグが立っているアルゴリズムがなければステップS51からの処理を繰り返す。

〔S53〕フラグが立っているアドレスに格納されたアルゴリズムで暗号化されているプログラム又はデータを、フラグが立っていないアドレスのアルゴリズムで暗号化し直す。

〔S54〕対応する全てのプログラムの再暗号化が行われると、フラグが立っているアドレスを使用していない旨をアルゴリズム供給手段156に通知する。

【0119】アルゴリズム供給手段と暗号化方式選択手段とが以上のような処理を繰り返し行っていることにより、使用期限が過ぎたアルゴリズムが随時検出され、そのアルゴリズムが書き換えられる。しかも、書き換えの対象となったアルゴリズムで暗号化されているプログラムを、他のアルゴリズムにより再暗号化した後に、そのアルゴリズムの書き換えが行われるため、書き換えを行ったことによりトラブルが発生することもない。

【0120】なお、書き換えフラグが立っているにもかかわらず、一定期間アルゴリズムの書き換えが行われなかった場合には、強制的にアルゴリズムを書き換えることもできる。

【0121】図30は強制的なアルゴリズムの書き換えの動作例を示すフローチャートである。この例では、書き換えフラグを立てて10分の間に必ず鍵の書き換えを行う。

〔S61〕使用期限となったアルゴリズム・アドレスがあるか否かを判断し、使用期限となったアルゴリズム・アドレスがあればステップS62へ進み、なければこのステップを繰り返す。

〔S62〕使用期限となったアルゴリズム・アドレスの書き換えフラグを立てる。

〔S63〕フラグが立っているアルゴリズム・アドレスのアルゴリズムが使用されているか否かを判断し、使用されていればステップS64に進み、使用されていなければステップS65に進む。

〔S64〕フラグが立てられてから10分が経過したか否かを判断し、経過していればステップS65に進み、経過していなければステップS63に進む。

〔S65〕使用期限となったアルゴリズム・アドレスのアルゴリズムを書き換え、ステップS61に進む。

【0122】このようにすることで、再暗号化手段が改ざんされて、アルゴリズムの変更を行わなくなってしまう場合でも、アルゴリズムは強制的に書き換えられ、その結果情報処理手段が正常動作しなくなることで、自己破壊的にデータの秘匿性を保つことができる。この例では10分の間に必ずアルゴリズムの書き換えを行うようにしたが、実際は、システムの使用状況や情報処理手段1の処理能力を勘案して決定するとよい。また、アルゴリズムを強制的に書き換える代わりにNMIを発生さ

せて、システムをリセットしてもよい。

【0123】以上の説明では、暗号／復号鍵を随時変更する場合と、暗号化／復号アルゴリズムを随時変更する場合とを個別に説明したが、鍵とアルゴリズムとの双方を随時変更することもできる。これらを組み合わせた構成にすることにより、いっそうの暗号強度の強化を図ることができる。

【0124】また、図23に示した構成を保護容器内に収めることにより、データの保護をさらに強化することができる。図31は主要部を保護容器に格納したソフトウェアの保護機能付き情報処理装置の例を示す図である。この例は、図23に示した構成に、単に保護容器150を加えたものであるため、同じものには同一の番号を付し説明を省略する。この保護容器150には、情報処理手段151、流通ソフト復号手段152、暗号化手段153、復号手段154、鍵供給手段155、アルゴリズム供給手段156、及び暗号化方式選択手段157が収納されている。保護容器150は、内部に収納した装置に対する外部からのプロービングに対して保護するような仕組みを有する容器である。保護容器150はそれを取り外されたり穴をあけられたりした時に内部の動作を停止させたり、秘匿データが読み出されないように消去する機能を有している。これにより、暗号化／復号アルゴリズム、暗号／復号鍵、及び復号されたデータの盗用や改ざんを完全に防ぐことができる。なお、保護容器の具体例は、図2において説明したものと同様である。

#### 【0125】

【発明の効果】以上説明したように本発明では、暗号化及び復号の際に、対象となるデータの属性に応じて暗号／復号鍵および暗号化／復号アルゴリズムを生成するようにしたため、データの属性ごとに異なる暗号／復号鍵および暗号化／復号アルゴリズムを使用し、メモリの管理方法に依存せずに暗号強度を強化することが可能となる。

#### 【図面の簡単な説明】

【図1】本発明の原理構成を示す図である。

【図2】ソフトウェアの保護機能付き情報処理装置のハードウェア構成例を示すブロック図である。

【図3】複数の復号手段を有するソフトウェアの保護機能付き情報処理装置のハードウェア構成を示すブロック図である。

【図4】図2と図3の情報処理装置を融合させたソフトウェアの保護機能付き情報処理装置のハードウェア構成を示すブロック図である。

【図5】鍵選択手段で鍵群を選択する例の詳細を示す図である。

【図6】再暗号化手段による鍵の変更方法の第1の例を示す図である。(A)は鍵の変更前の状態を示す図であり、(B)は鍵の変更指令出力後の状態を示す図であ

り、(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。

【図7】再暗号化手段による鍵の変更方法の第2の例を示す図である。(A)は鍵の変更前の状態を示す図であり、(B)は鍵の変更指令出力後の状態を示す図であり、(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。

【図8】鍵供給手段内の鍵を変更する場合の例を示す図である。(A)は鍵の変更前の状態を示す図であり、(B)は鍵の変更指令出力後の状態を示す図であり、(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。

【図9】鍵の生成方式を示す図である。(A)は新たな鍵を作成する方式を示す図であり、(B)は鍵テーブルを利用する方式を示す図である。

【図10】識別子を用いた鍵の選択方式を示す図である。(A)は鍵の変更前の状態を示す図であり、(B)は鍵の変更後の状態を示す図である。

【図11】時間管理テーブルを示す図である。

【図12】時間管理手段の処理手順を示すフローチャートである。

【図13】再暗号化手段の処理手順を示すフローチャートである。

【図14】時間管理手段の鍵書き換えに関する他の動作例を示すフローチャートである。

【図15】ページ番号と鍵識別子の対応を管理する鍵管理テーブルの例を示す図である。

【図16】鍵選択手段内のTLBを示す図である。

【図17】MMU内のTLBを示す図である。

【図18】インテル社製のMPUであるi80286のタスク状態セグメントに鍵アドレスの情報を付加した例を示す図である。

【図19】データの内容による鍵管理の暗号化の処理手順を示すフローチャートである。

【図20】図19の各ステップの状態遷移図である。

(A)はステップS21の状態を示す図であり、(B)はステップS22の状態を示す図であり、(C)はステップS23の状態を示す図であり、(D)はステップS24の状態を示す図であり、(E)は、ステップS25の状態を示す図である。

【図21】データの内容による鍵管理の復号の処理手順を示すフローチャートである。

【図22】図21の各ステップの状態遷移図である。

(A)はステップS31の状態を示す図であり、(B)はステップS32の状態を示す図であり、(C)はステップS33の状態を示す図であり、(D)はステップS34の状態を示す図であり、(E)は、ステップS35の状態を示す図である。

【図23】暗号化／復号アルゴリズムを随時変更するソフトウェア保護機能付き情報処理装置のハードウェア構

成例を示すブロック図である。

【図24】暗号化方式の変更例を示す図である。(A)はアルゴリズムの変更前の状態を示す図であり、(B)はアルゴリズムの変更指令出力後の状態を示す図であり、(C)は再暗号化が必要な全てのデータの再暗号化が終了した後の状態を示す図である。

【図25】暗号化方式を変更する際に鍵群を選択する例を示す図である。

【図26】供給するアルゴリズムを変更する際の処理手順を示す図である。(A)はアルゴリズムの変更前の状態を示す図であり、(B)はアルゴリズムの変更指令出力後の状態を示す図であり、(C)はアルゴリズムを変更後の状態を示す図である。

【図27】アルゴリズムの使用時間の経過時にアルゴリズムの変更を行う例を示す図である。

【図28】アルゴリズム供給手段のアルゴリズムの書き換え処理のフローチャートである。

【図29】暗号化方式選択手段のアルゴリズム切り換え処理のフローチャートである。

【図30】強制的なアルゴリズムの書き換え処理の動作例を示すフローチャートである。

【図31】主要部を保護容器に格納したソフトウェアの保護機能付き情報処理装置の例を示す図である。

【図32】従来のソフトウェアの保護を図った情報処理装置のブロック図である。

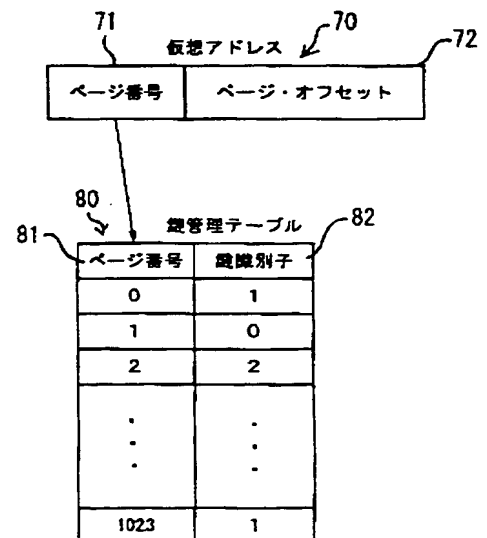
# 【符号の説明】

- 1 暗号鍵生成手段
- 1a 暗号鍵
- 2 復号鍵生成手段
- 2a 復号鍵
- 3 暗号化手段
- 4 記憶手段
- 4a 暗号データ
- 5 復号手段
- 6 処理手段
- 7 制御手段
- 10 安全保護容器
- 11 情報処理手段
- 12 暗号化手段
- 13 復号手段
- 14 鍵選択手段
- 15 鍵供給手段
- 20 システムバス
- 21 記憶手段
- 22 I/Oインターフェース
- 23 ペリフェラルバス
- 24 イーサネット・コントローラ
- 25 CD-ROMドライブ
- 26 ハードディスク

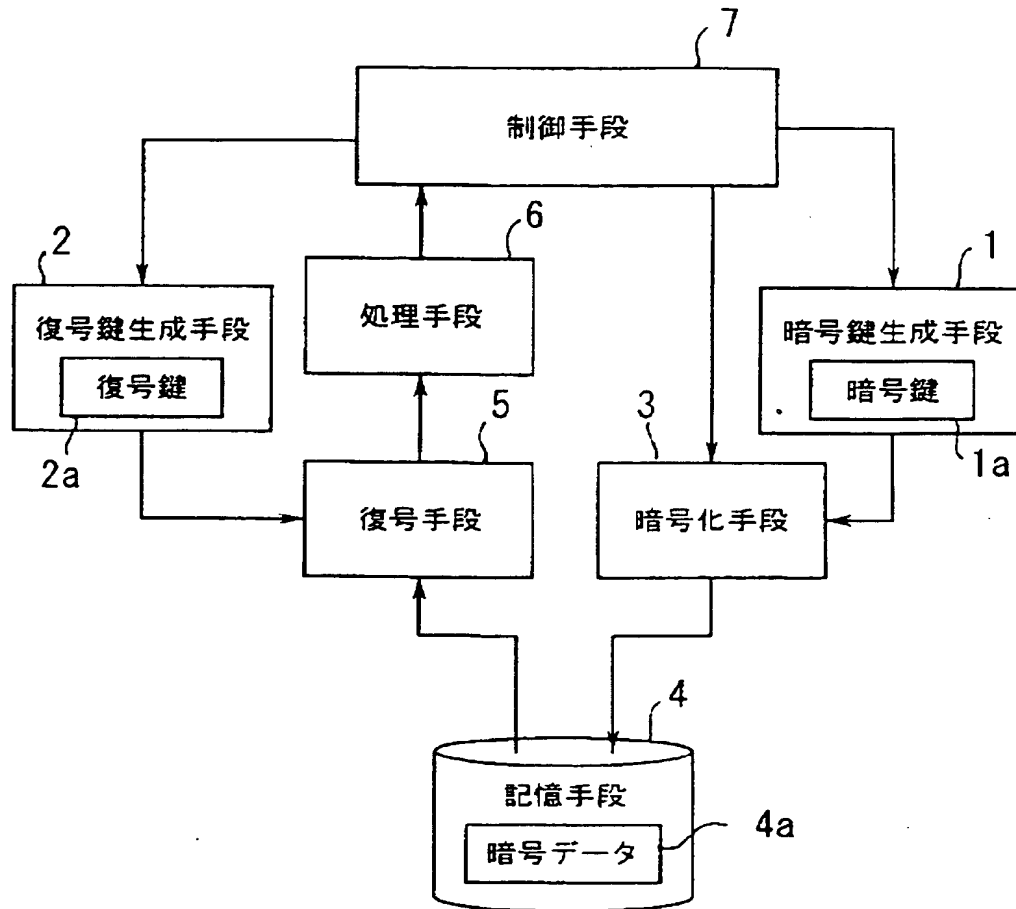
【図11】

59a 鍵識別子	59b 鍵寿命	59c 経過時間	59d 書き換えフラグ
0	10分	1分	0
1	60分	30分	0
2	60分	61分	1

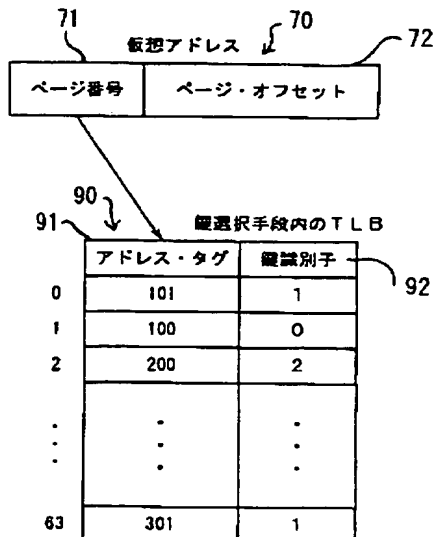
【図15】



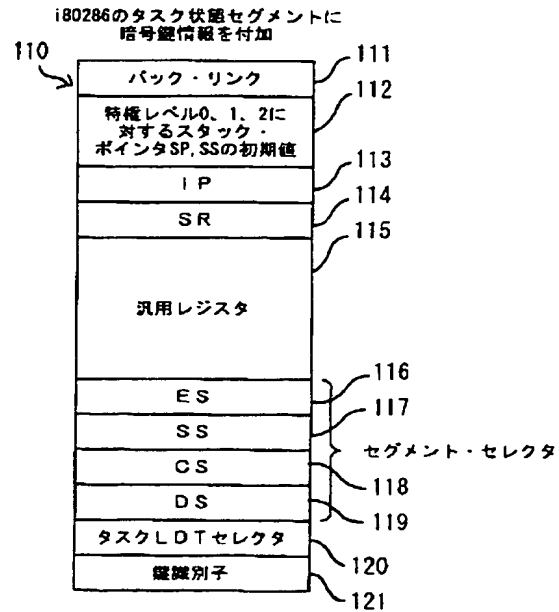
【図1】



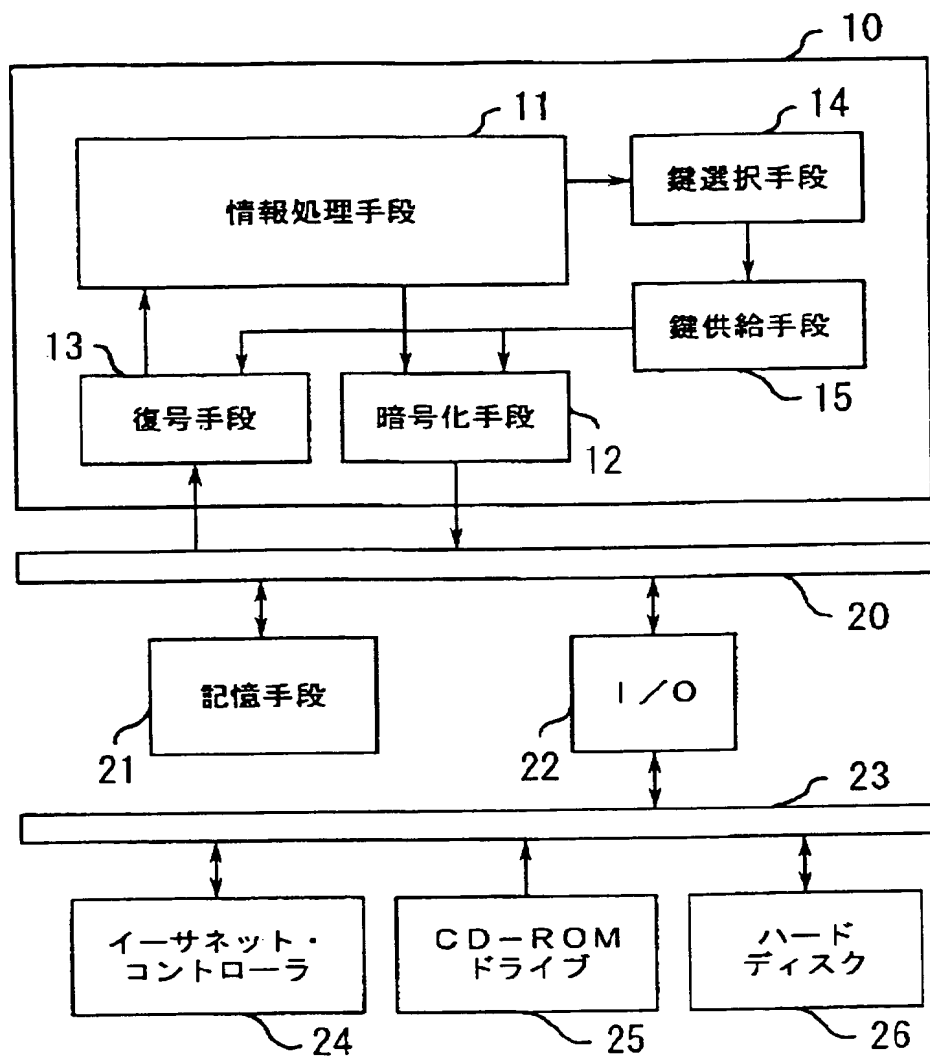
【図16】



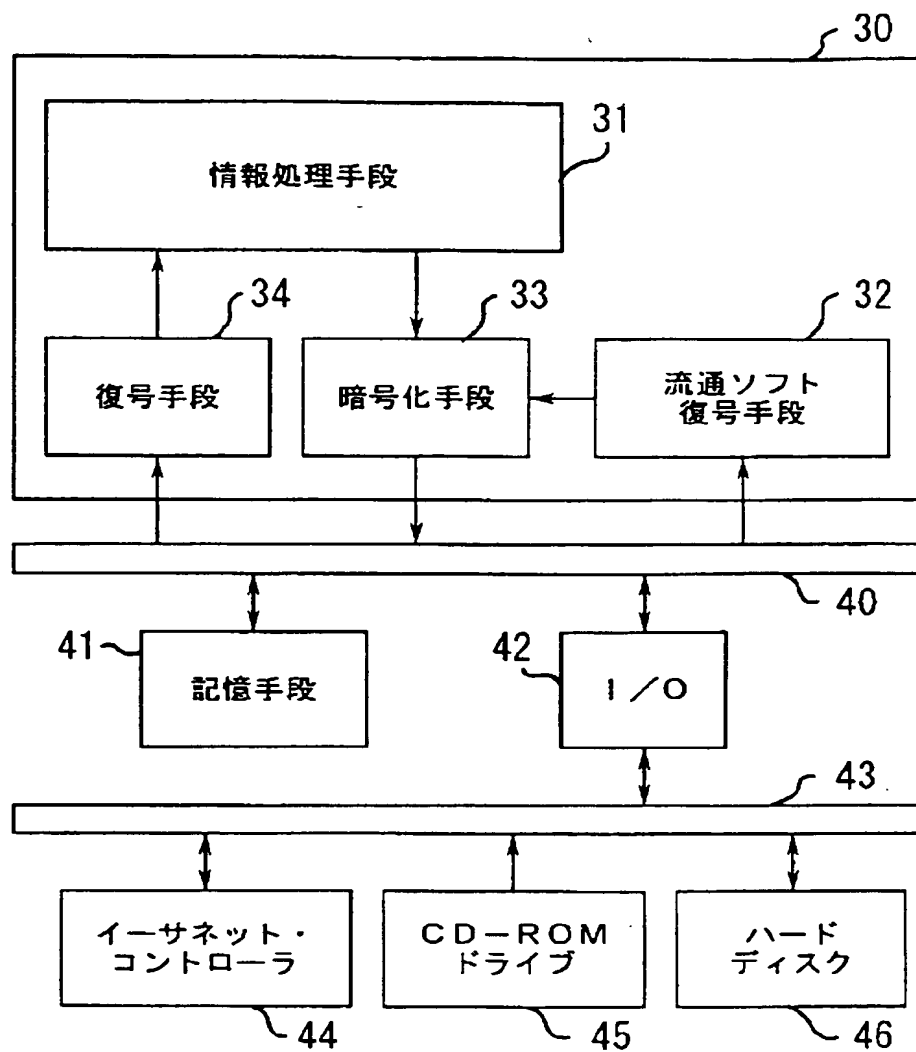
【図18】



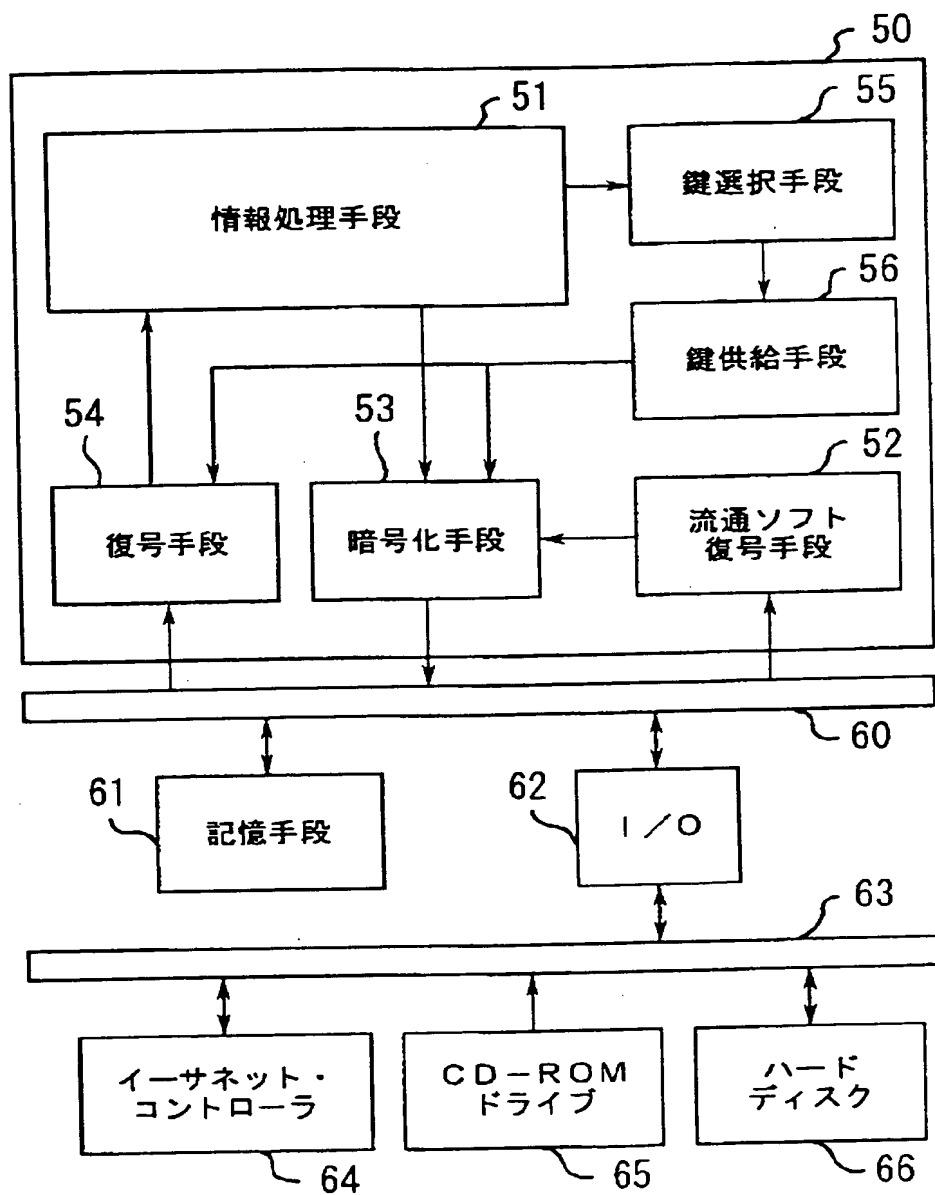
【図2】



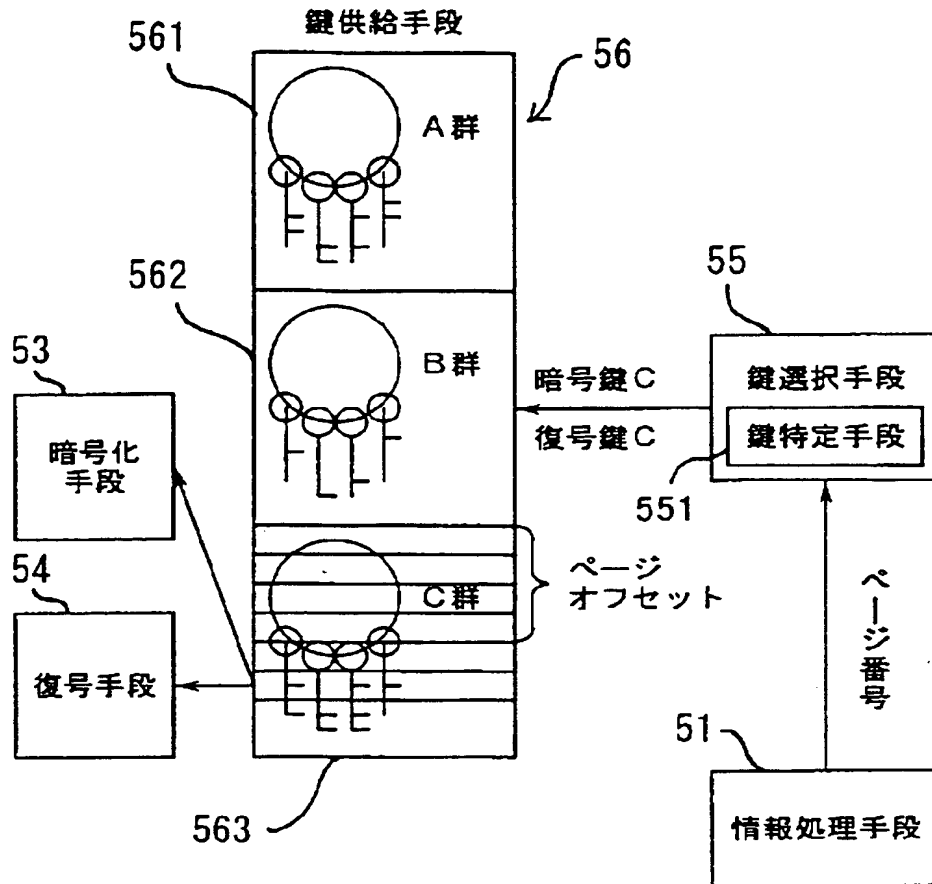
【図3】



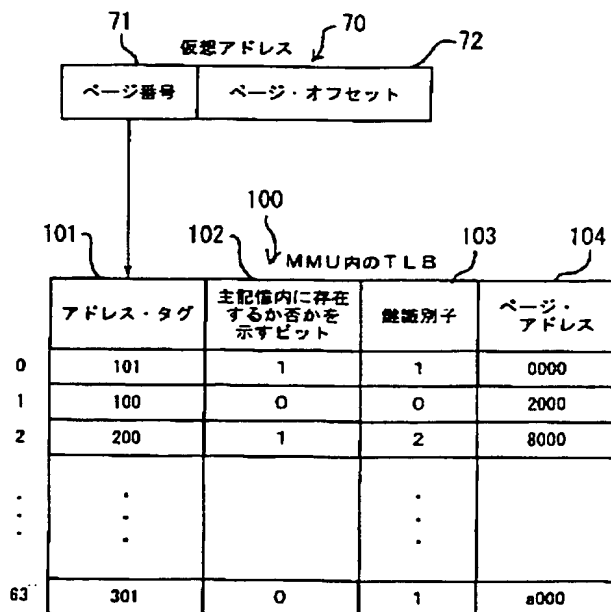
【図4】



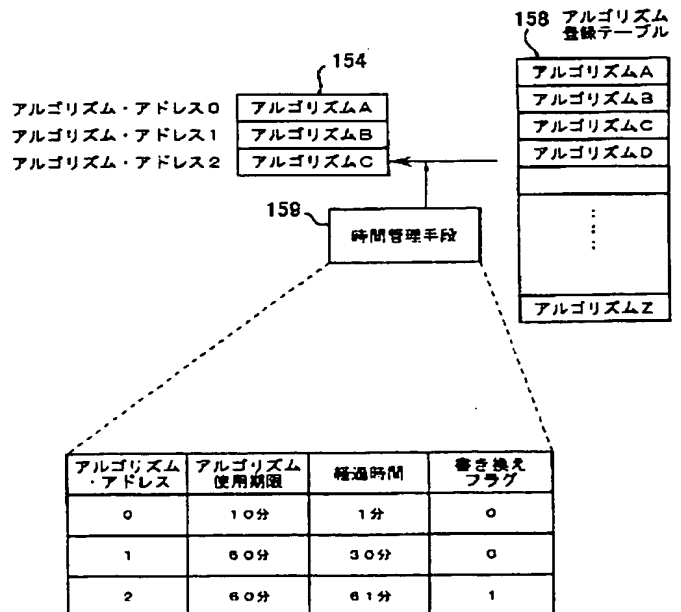
【図5】



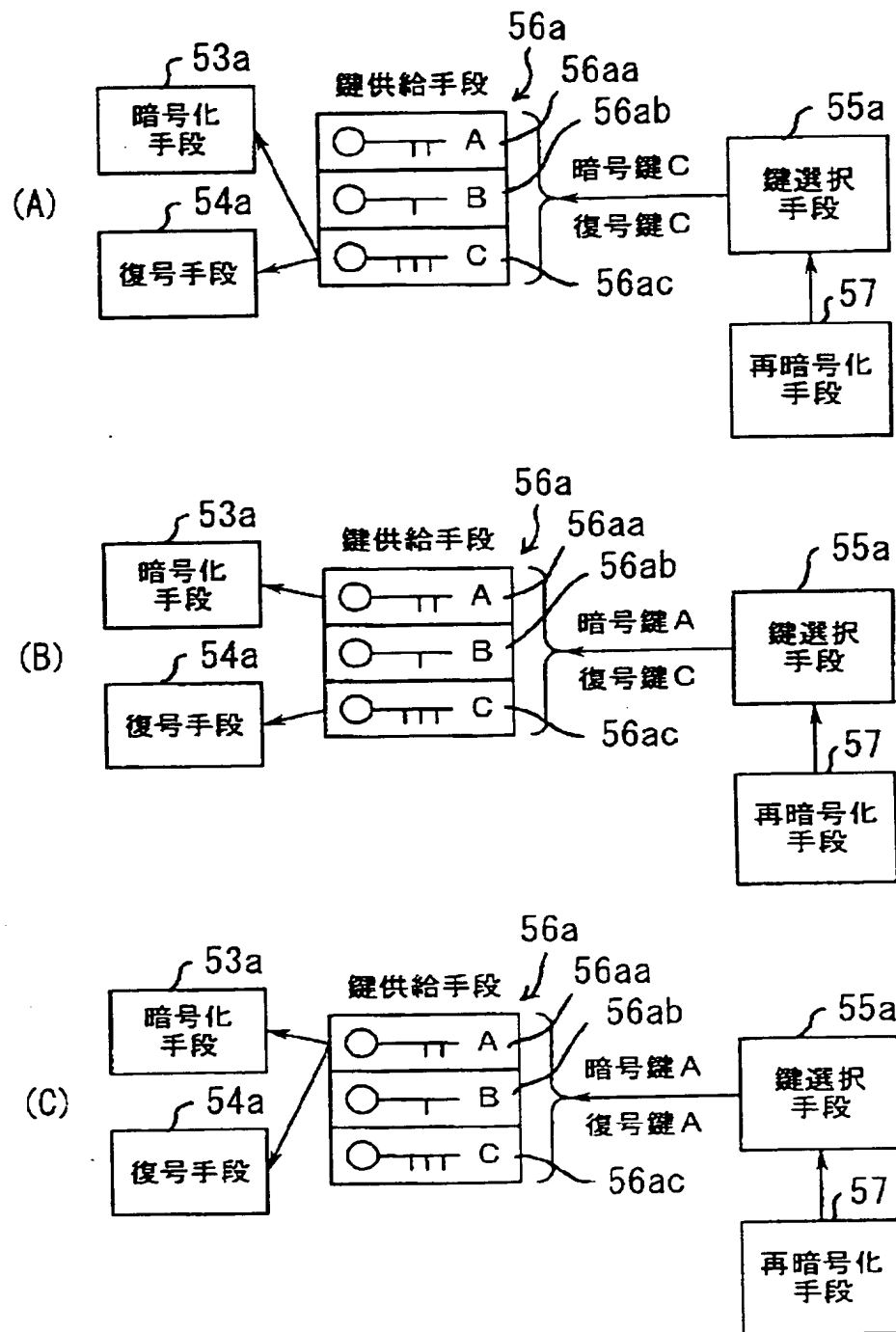
【図17】



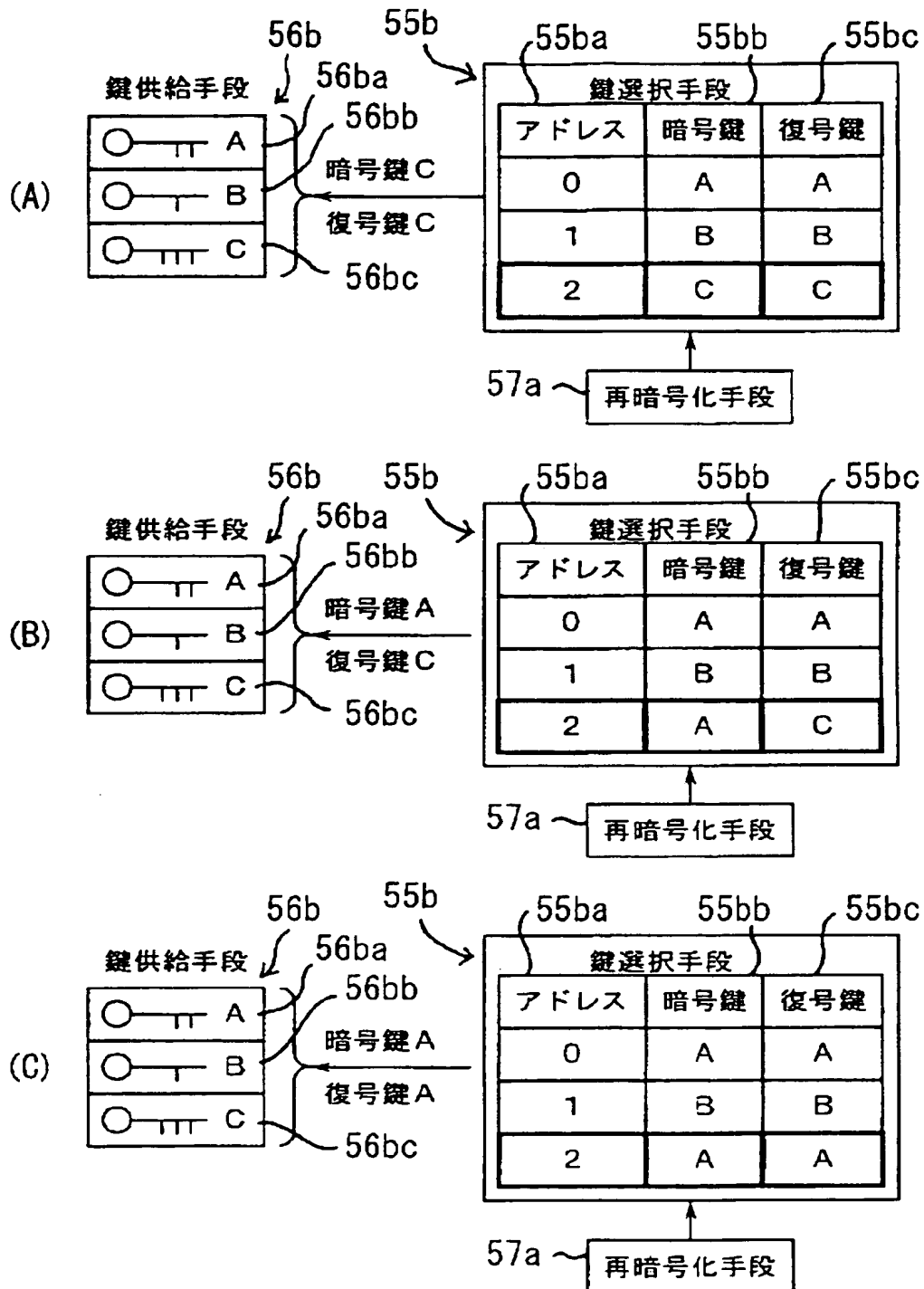
【図27】



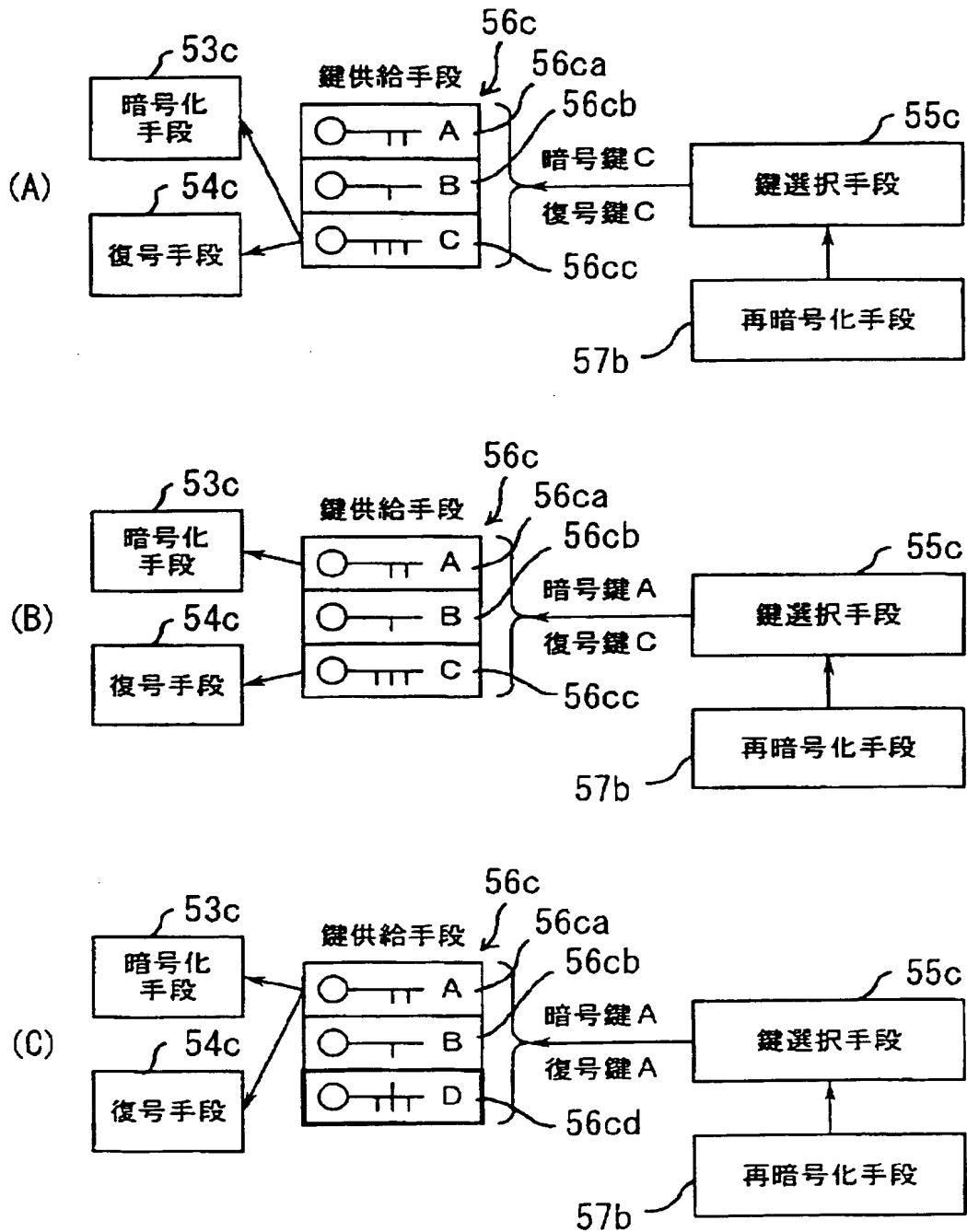
【図6】



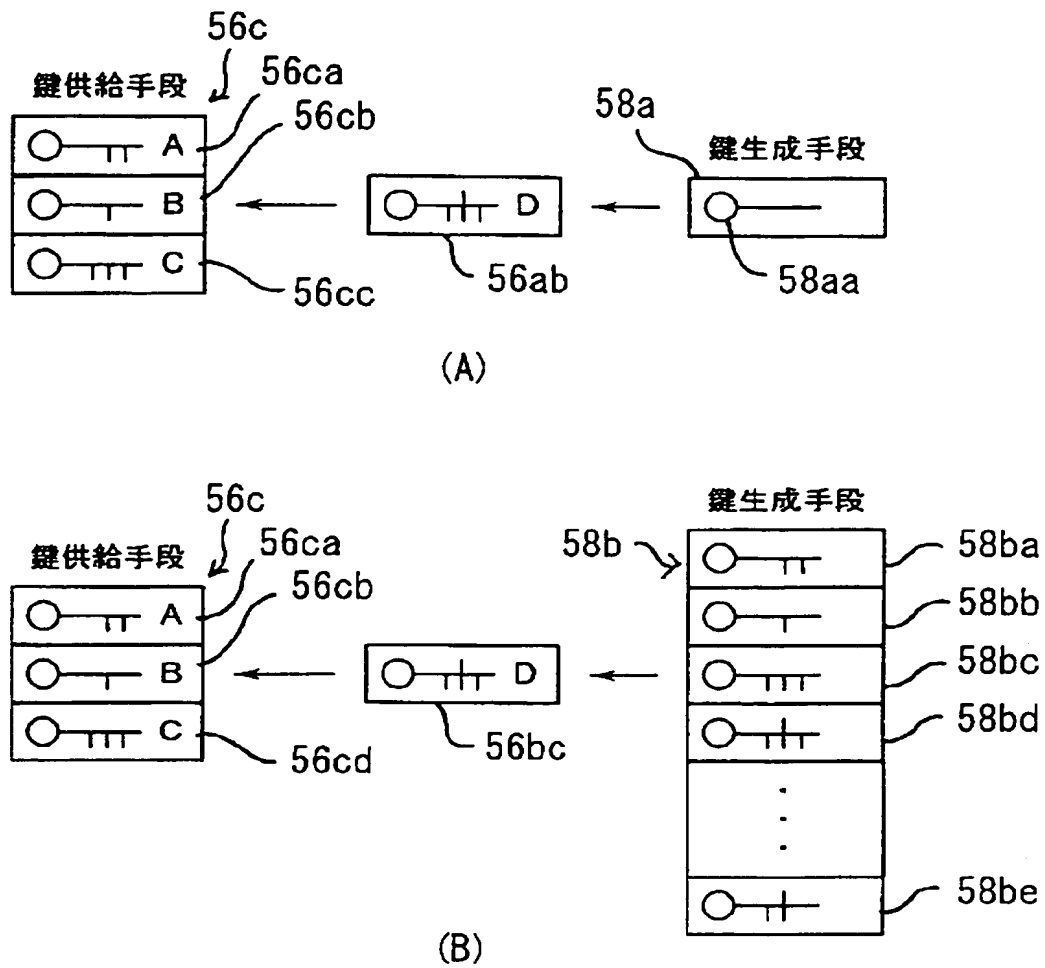
【図7】



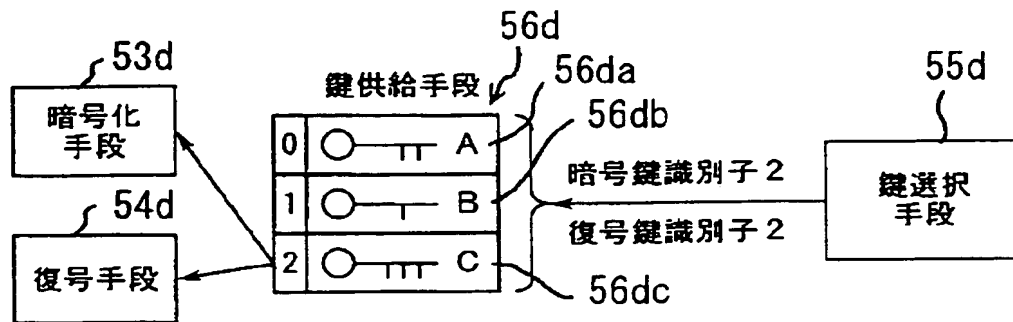
【図8】



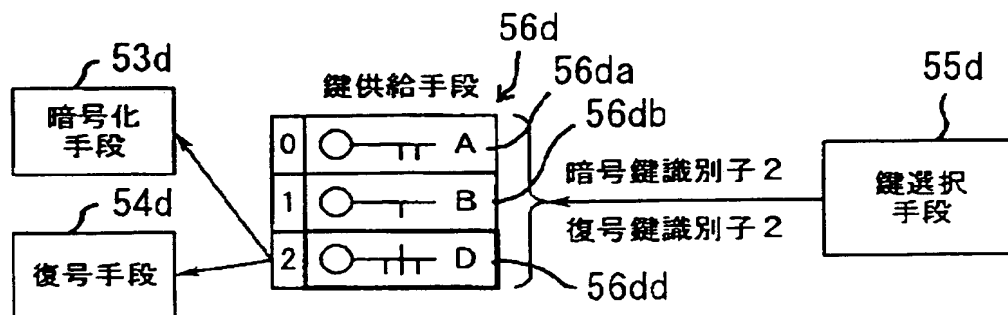
【図9】



【図10】

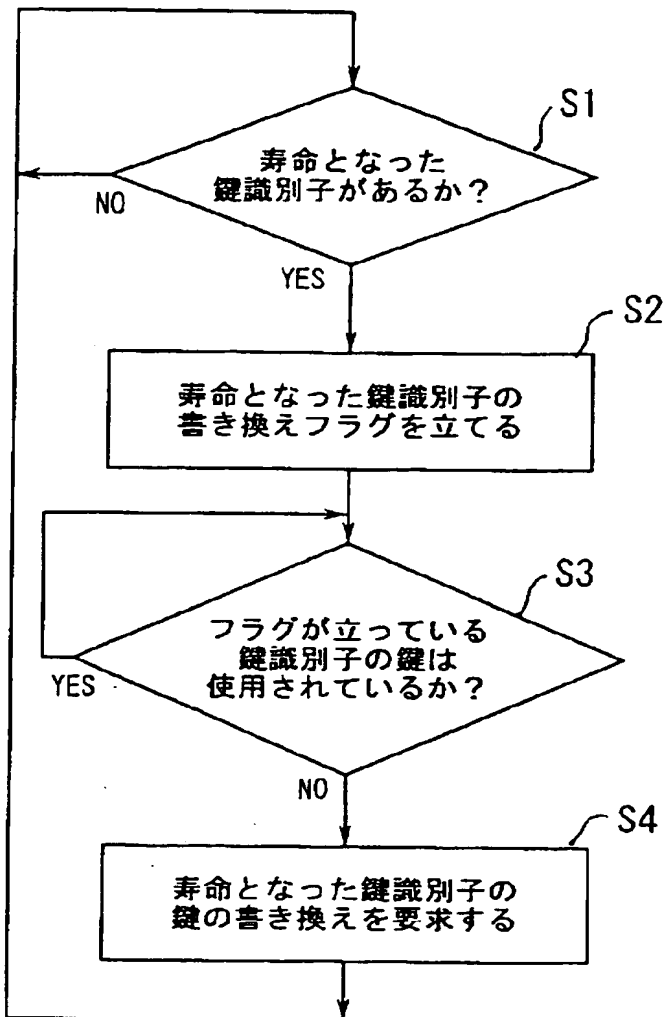


(A)

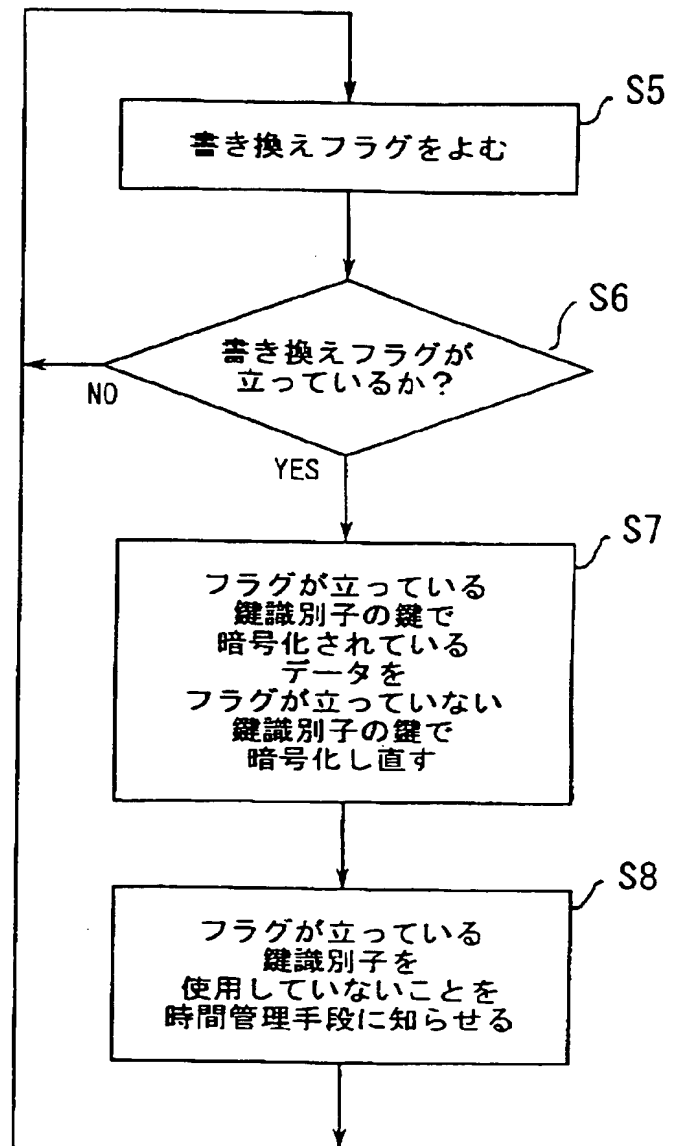


(B)

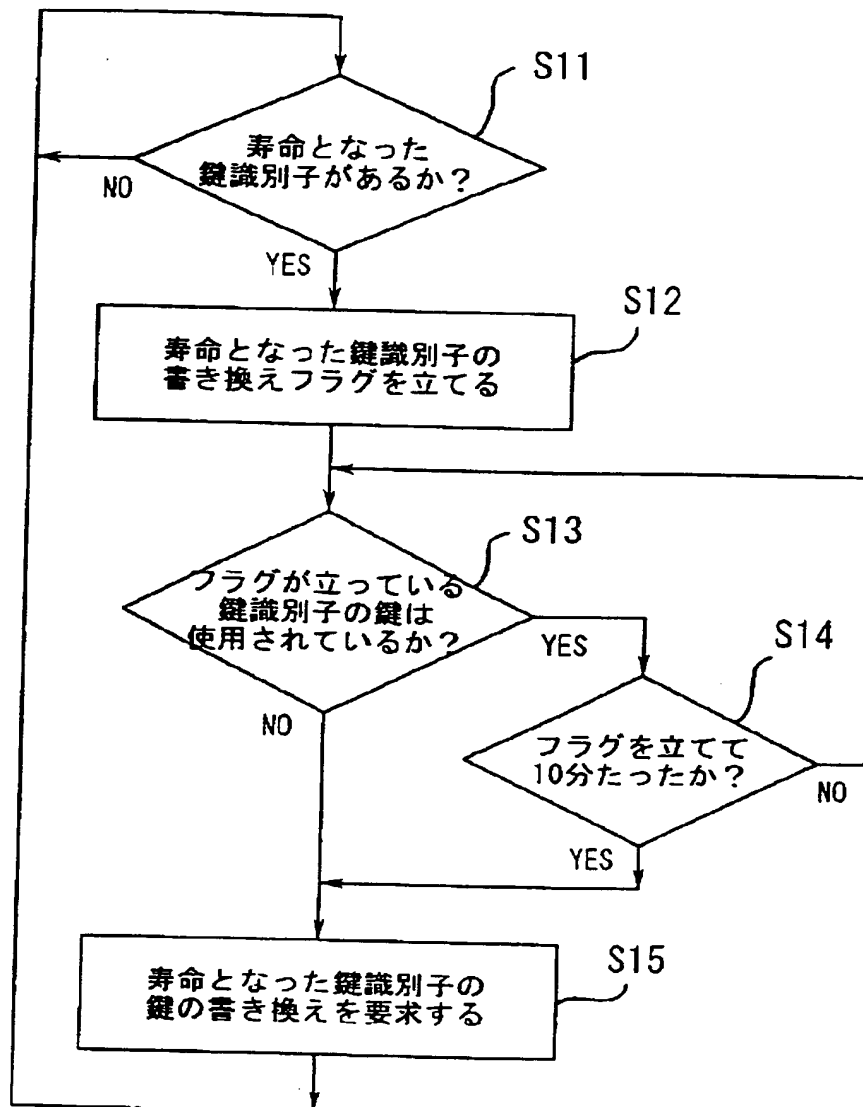
【図12】



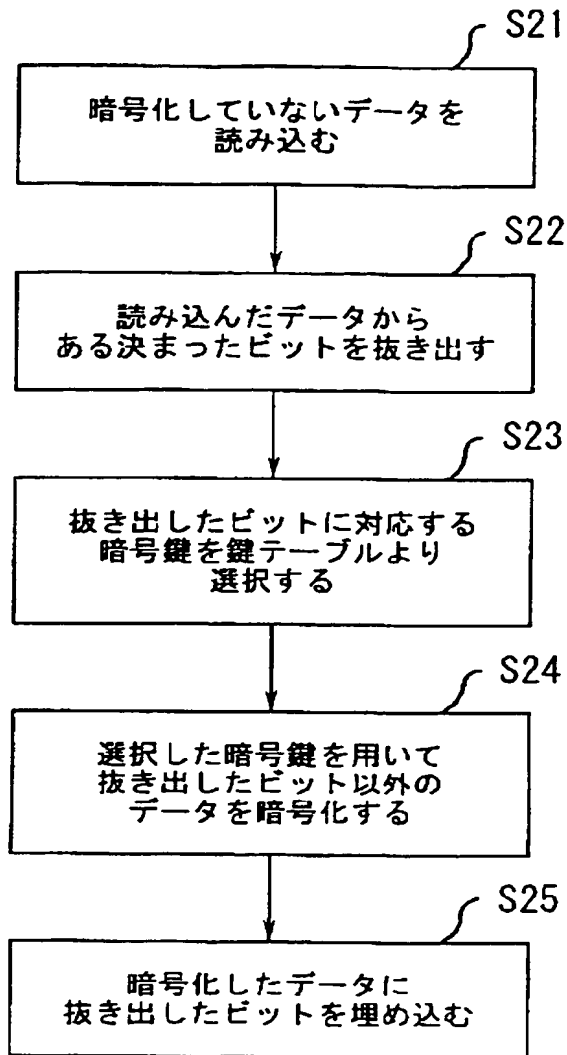
【図13】



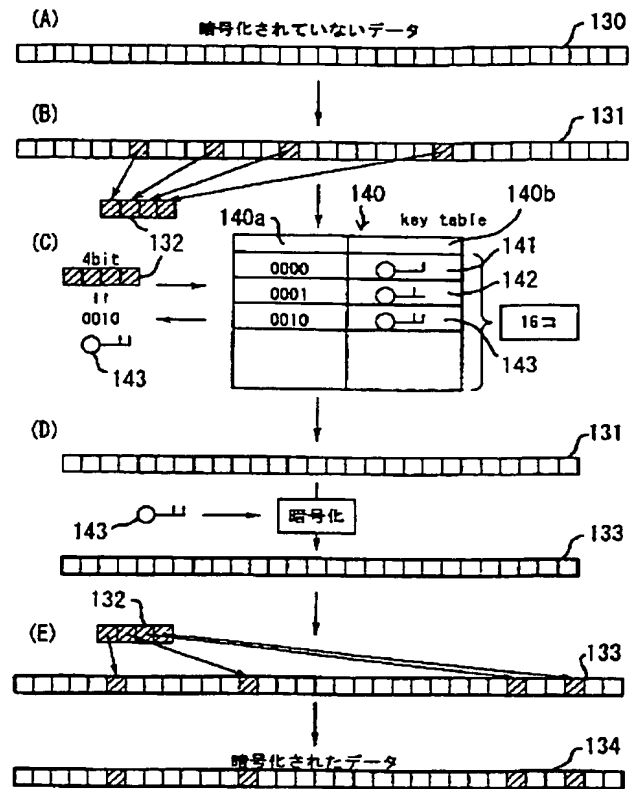
【図14】



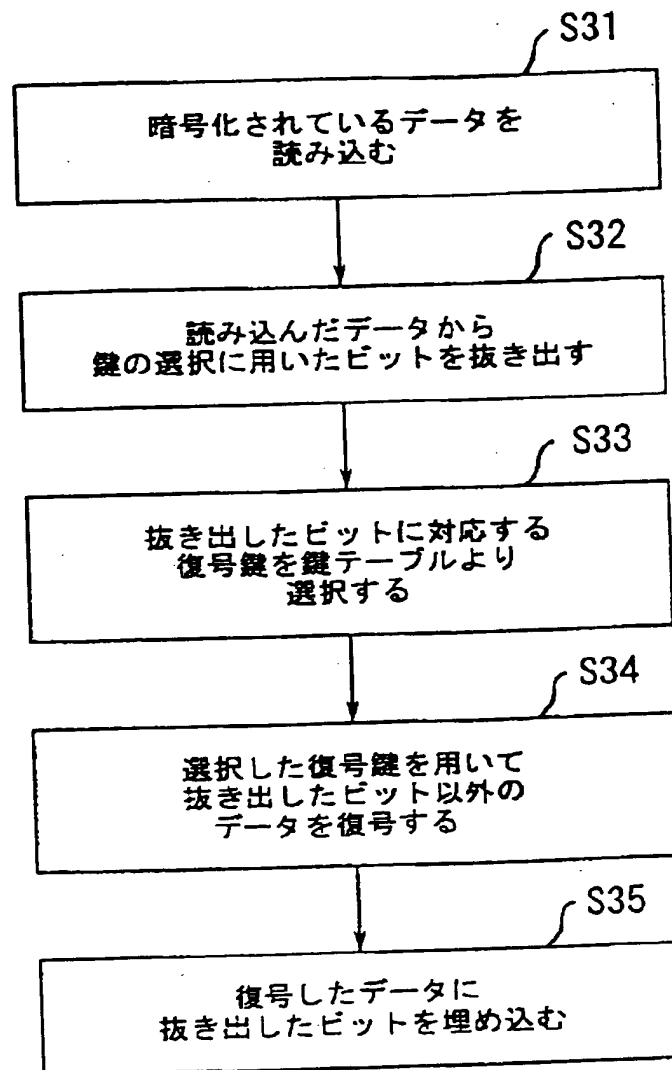
【図19】



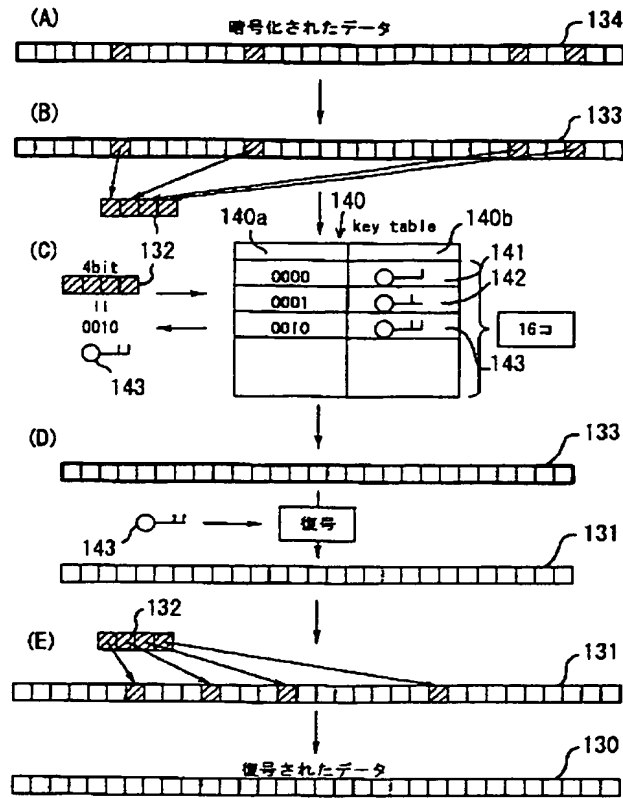
【図20】



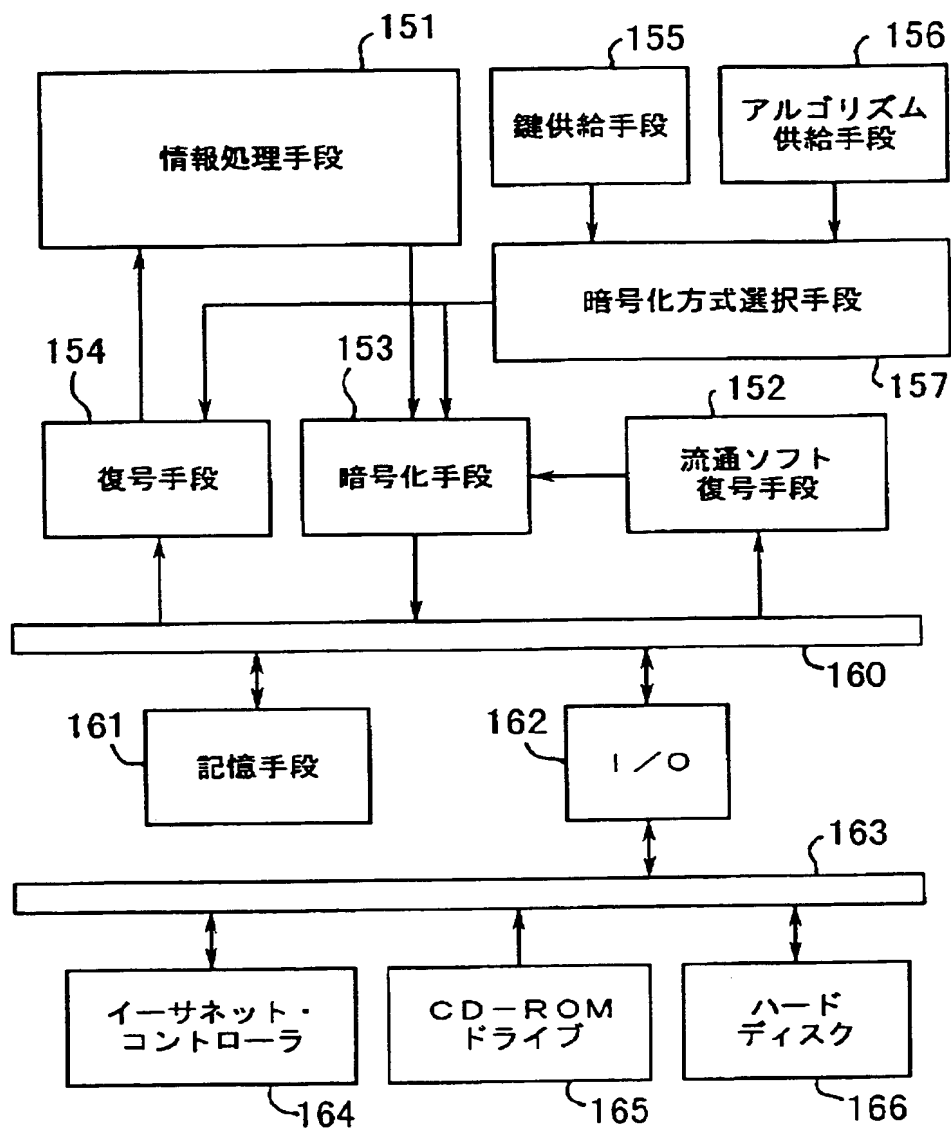
【図21】



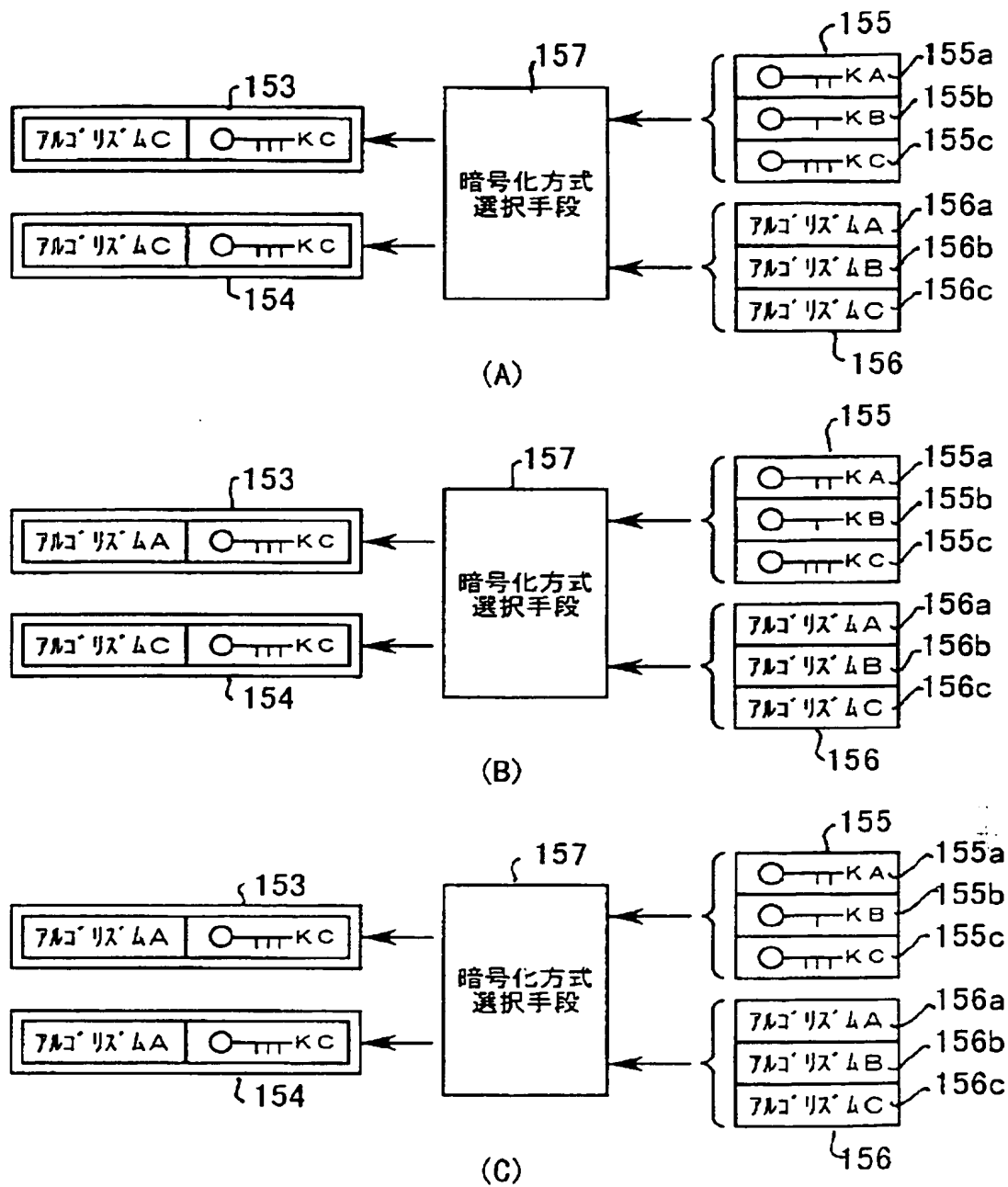
【図22】



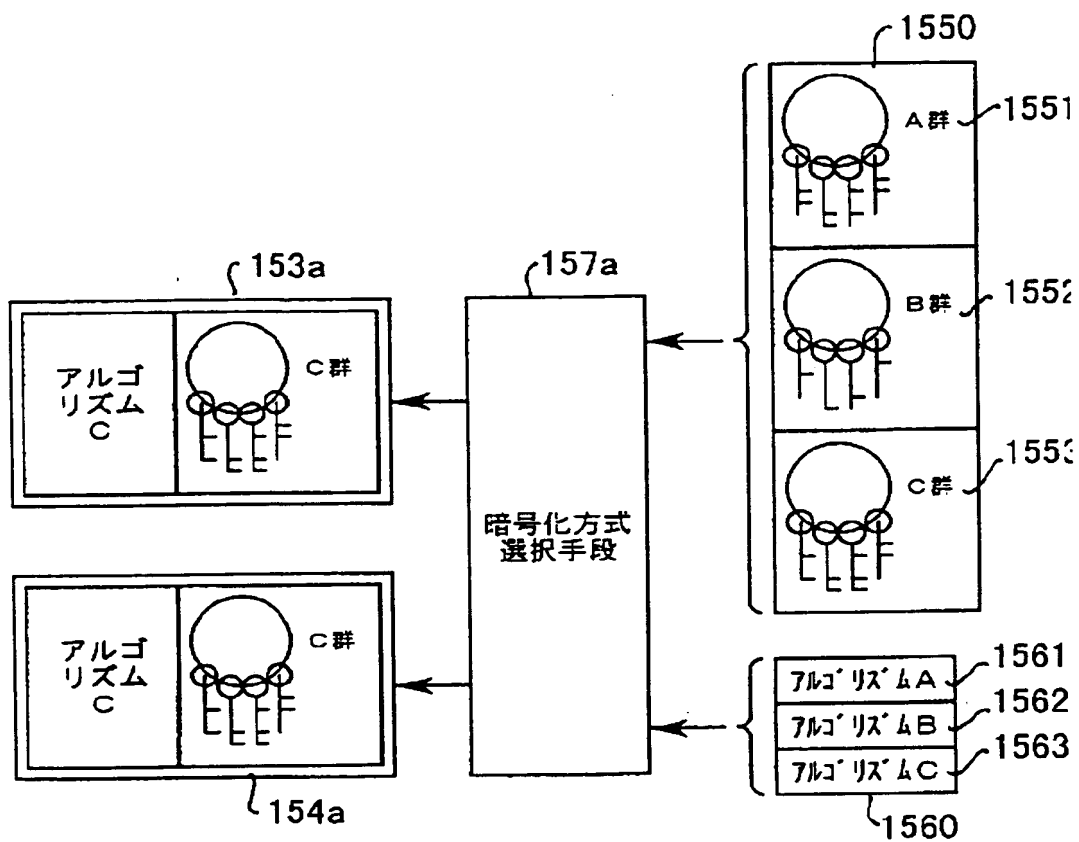
【図23】



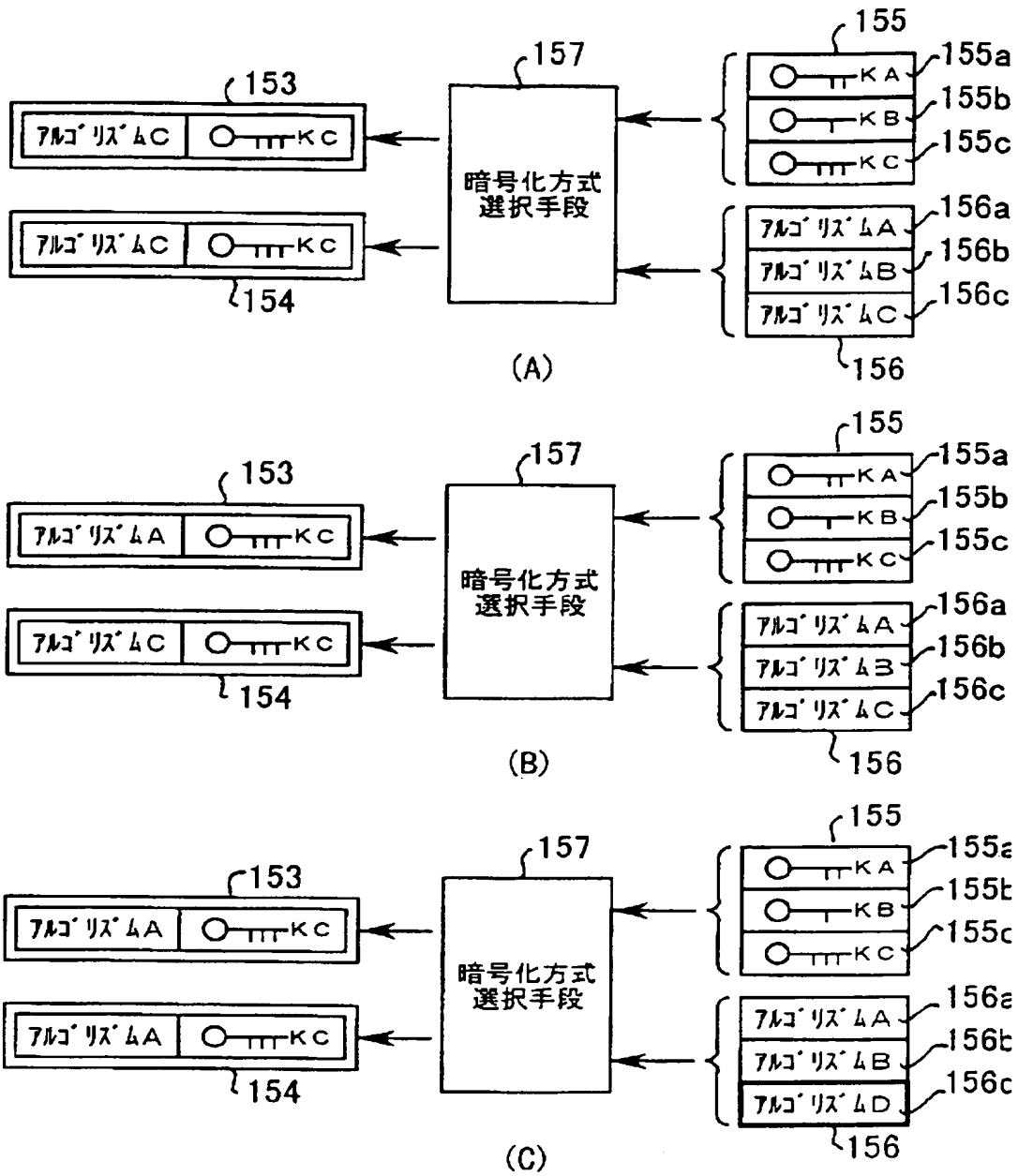
【図24】



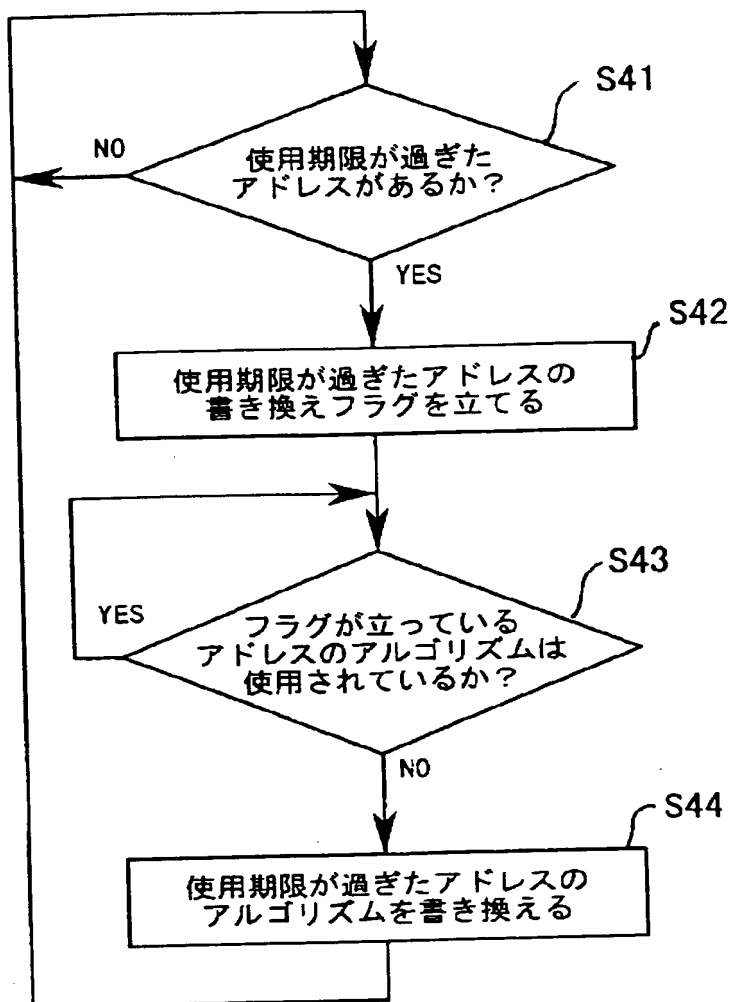
【図25】



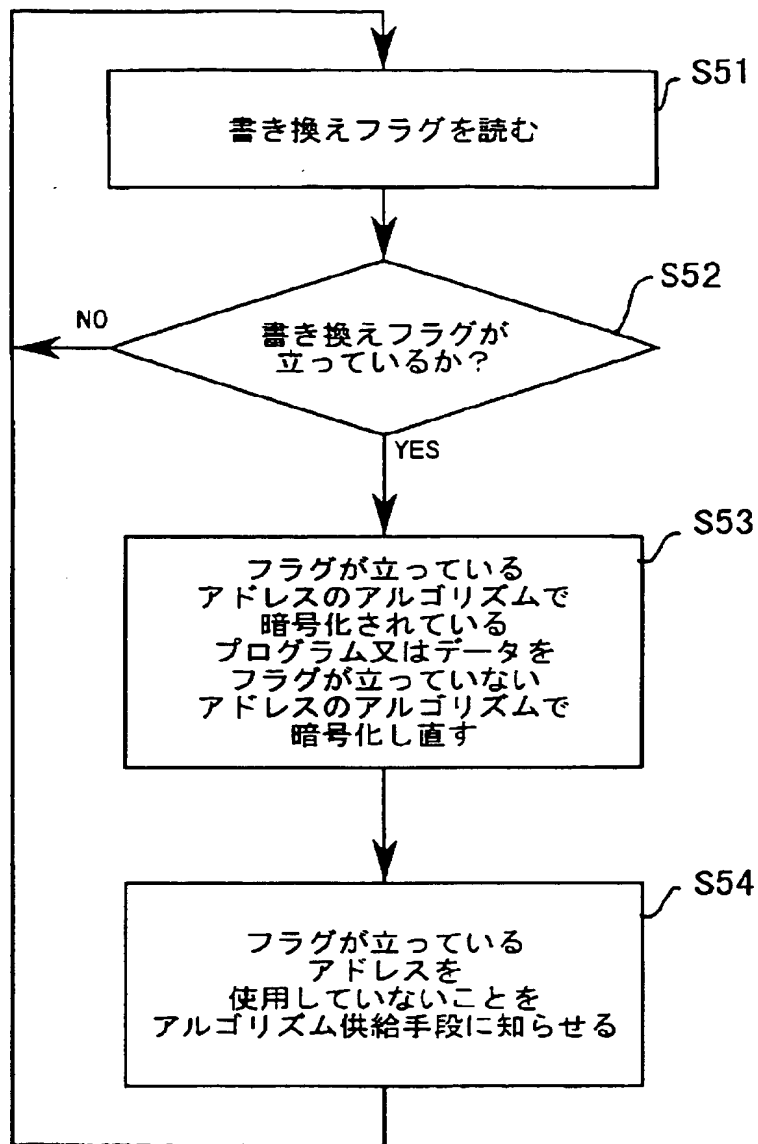
【図26】



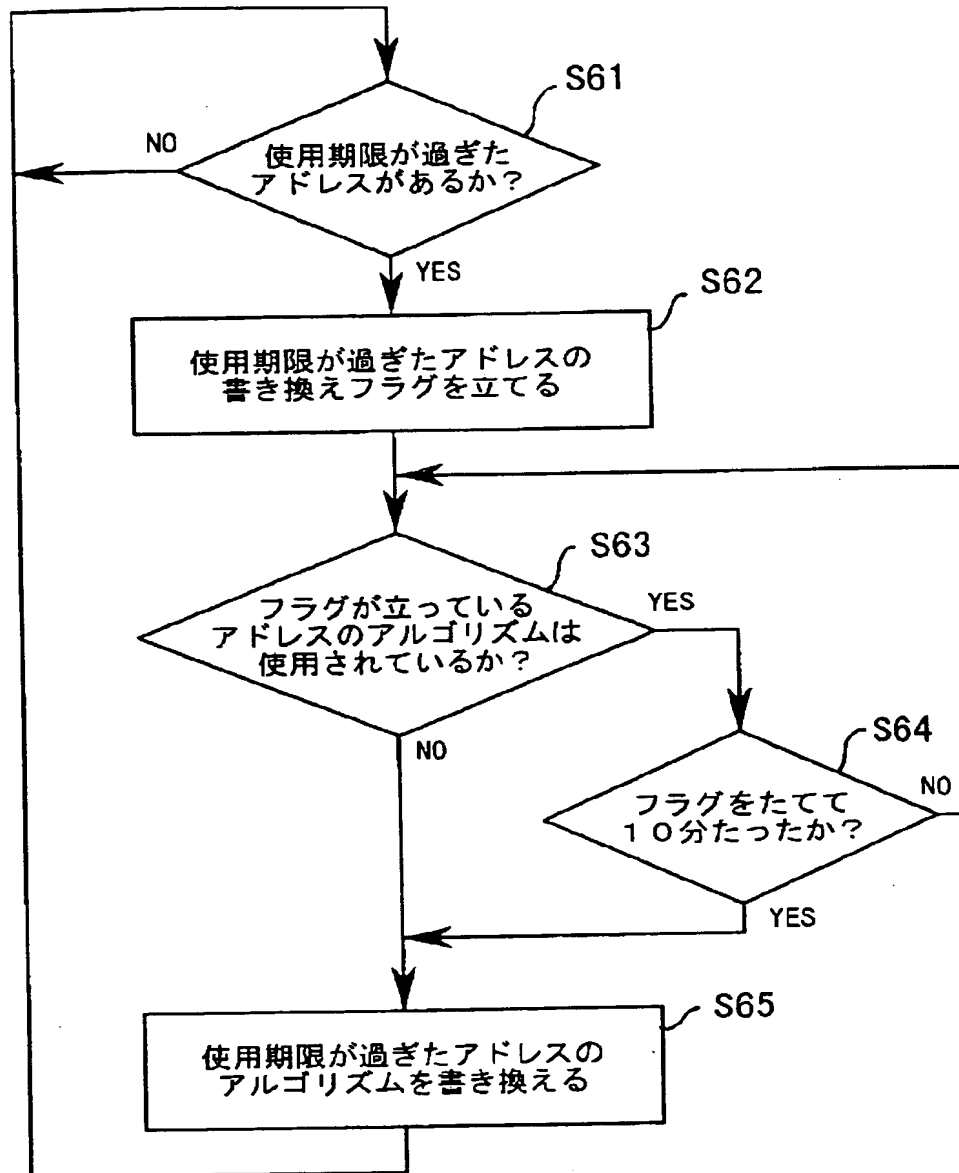
【図28】



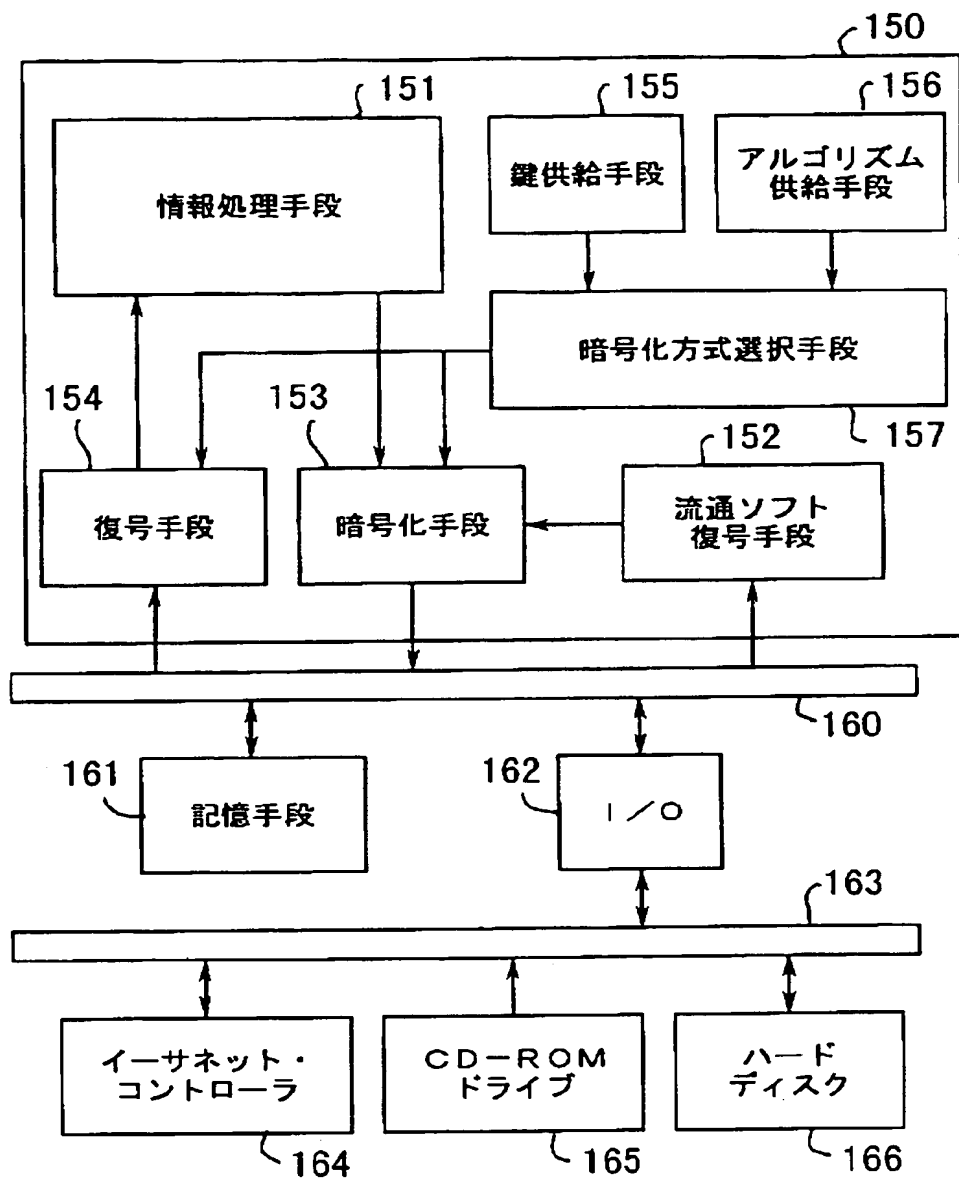
【図29】



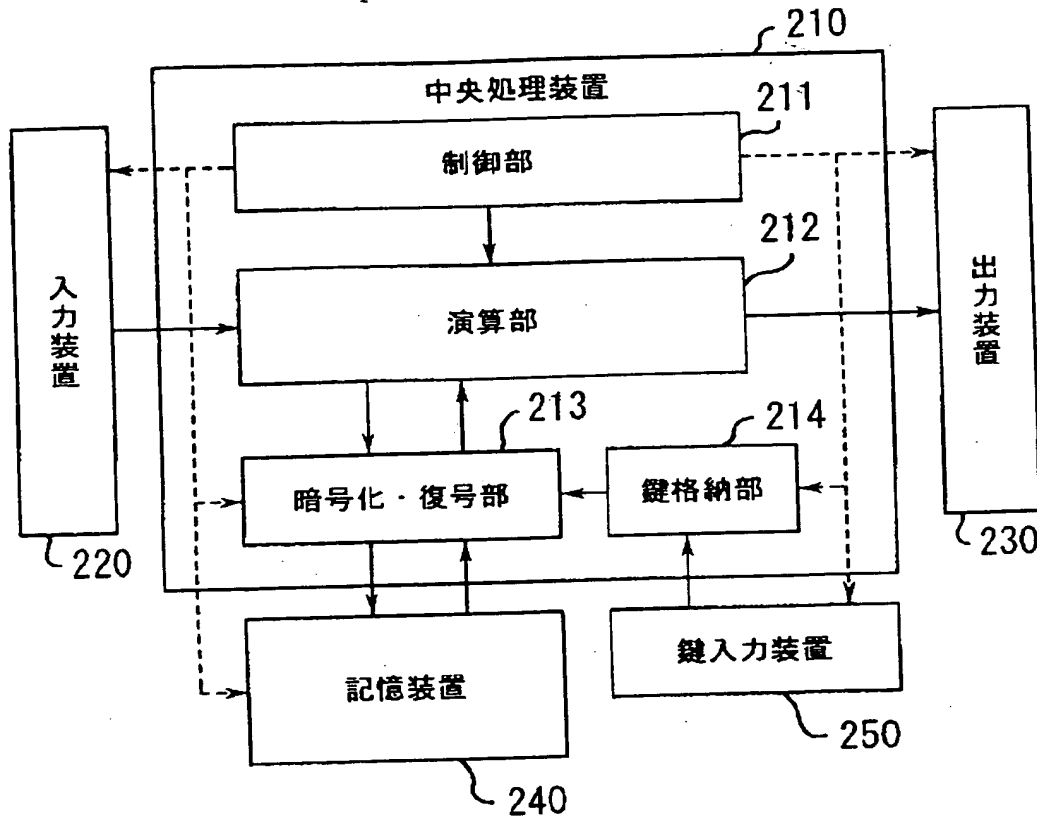
【図30】



【図31】



【図32】



フロントページの続き

(51) Int. Cl.<sup>6</sup>  
H04L 9/08  
9/14

識別記号 庁内整理番号

FI  
H04L 9/00

技術表示箇所

601B  
641